

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ
АВТОМАТИЗОВАНИХ СИСТЕМ

КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Розробка IoT- системи для автоматичного управління побутовими
електроприладами»

на здобуття освітнього ступеня бакалавра
зі спеціальності 126 Інформаційні системи та технології
(код, найменування спеціальності)
освітньо-професійної програми Інформаційні системи та технології
(назва)

*Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають
посилання на відповідне джерело*

(підпис)

Владислав СТРЕЛЬЧЕНКО
Ім'я, ПРИЗВИЩЕ здобувача

Виконав: здобувач вищої освіти гр. ІСД-42

Владислав СТРЕЛЬЧЕНКО
Ім'я, ПРИЗВИЩЕ

Керівник: Анастасія КАЗНАЧЕСЬ

Науковий ступінь, Ім'я, ПРИЗВИЩЕ
вчене звання

Рецензент: _____

Науковий ступінь Ім'я ПРИЗВИЩЕ
вчене звання

Київ 2024

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

Навчально-науковий інститут Інформаційних технологій

Кафедра Інженерії програмного забезпечення автоматизованих систем

Ступінь вищої освіти Бакалавр

Спеціальність Інформаційні системи та технології

Освітньо-професійна програма Інформаційні системи та технології

ЗАТВЕРДЖУЮ

Завідувач кафедрою ПЗАС

_____ Каміла СТОРЧАК

« ____ » _____ 2024 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**
Стрельченку Владиславу Сергійовичу
(прізвище, ім'я, по-батькові здобувача)

1. Тема кваліфікаційної роботи: Розробка IoT- системи для автоматичного управління побутовими електроприладами

керівник кваліфікаційної роботи Анастасія КАЗНАЧЕСЬКА

(Ім'я, ПРИЗВИЩЕ, науковий ступінь, вчене звання)

затверджені наказом Державного університету інформаційно-комунікаційних технологій від "27" лютого 2024 р. №36

2. Строк подання кваліфікаційної роботи «31» травня 2024 р.

3. Вхідні дані до кваліфікаційної роботи:

1. Інформаційно-ентропійний метод;
2. Sustainability in New and Emerging Technologies;
3. Документація ESP8266;
4. Wireless Communication Technologies used in a Smart Home

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити).

1. Огляд систем автоматизації розумного дому
2. Аналіз теоретичних основ розробки та експлуатації систем сонячної електроенергетики
3. Опис реалізації апаратної частини системи

4. Опис розробки програмного забезпечення системи

5. Ілюстративний матеріал: *презентація*

6. Дата видачі завдання 27 лютого 2024 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Термін виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	14.04-18.04.2024	
2	Огляд особливостей функціонування автоматизованих систем розумного будинку	19.04.-22.04.2024	
3	Вивчення технологічної бази розробки системи	23.04.-25.04.2024	
4	Розробка концепції апаратної частини системи	26.04-29.04.2024	
5	Розробка програмного забезпечення системи	30.04-3.05.2024	
6	Написання основної частини кваліфікаційної роботи	4.05.- 18.05.2024	
7	Написання вступу, висновків та реферату	19.05.-22.05.2024	
8	Розробка демонстраційних матеріалів	23.05.-24.05.2024	

Здобувач(ка) вищої освіти

(підпис)

Владислав СТРЕЛЬЧЕНКО

(Ім'я, ПРИЗВІЩЕ)

Керівник

кваліфікаційної роботи.

(підпис)

Анастасія КАЗНАЧЕСЬКА

(Ім'я, ПРИЗВІЩЕ)

РЕФЕРАТ

Текстова частина бакалаврської роботи: 70 с., 47 рис., 5 табл., 4 схеми, 15 джерел.

Мета роботи – дослідження стану сучасних технологій в області IoT, розробка концепції системи автоматичного управління побутовими електроприладами.

Об'єкт дослідження – процес автоматизації управління побутовими електроприладами

Предмет дослідження – способи розробки та використання IoT-систем для автоматичного управління побутовими електроприладами

Короткий зміст роботи: в роботі реалізовано прототип системи автоматичного управління освітленням для догляду за рослинами на основі мікроконтролера NodeMCU ESP8266. Система включає датчики освітленості (BH1750) та руху (HC-SR501), які забезпечують автоматичне ввімкнення та вимкнення світлодіодної стрічки для підсвічування рослин залежно від умов навколишнього середовища та активності користувачів.

КЛЮЧОВІ СЛОВА: АВТОМАТИЗАЦІЯ РОЗУМНОГО БУДИНКУ, ІОТ, ЕНЕРГОЕФЕКТИВНІСТЬ, ESP8266, NODEMCU 8266, JavaScript, BH1750, HC-SR501.

ABSTRACT

Text Part of the Bachelor Thesis: 70 pages, 47 figures, 5 tables, 4 diagrams, 15 sources.

Purpose of the Work - To study the state of modern technologies in the field of IoT and to develop a concept for a system of automatic control of household electrical appliances.

Object of Study - The process of automating the control of household electrical appliances.

Subject of Study - Methods of developing and using IoT systems for automatic control of household electrical appliances.

Summar of the worky: the work implements a prototype system for automatic lighting control for plant care based on the NodeMCU ESP8266 microcontroller. The system includes light sensors (BH1750) and motion sensors (HC-SR501), which ensure automatic switching on and off of the LED strip for plant lighting depending on environmental conditions and user activity.

KEYWORDS: SMART HOME AUTOMATION, IOT, ENERGY EFFICIENCY, ESP8266, NODEMCU 8266, JavaScript, BH1750, HC-SR501.

ЗМІСТ

ВСТУП	9
1 ОГЛЯД ОСОБЛИВОСТЕЙ ФУНКЦІОНУВАННЯ СИСТЕМ РОЗУМНОГО БУДИНКУ	12
1.1 Сучасні системи управління електроприладами у розумному будинку	12
1.2 Дослідження ефективності автоматичного управління електроприладами для оптимізації використання електроенергії	23
1.3 Переваги та недоліки застосування популярних протоколів передачі даних у розумних будинках	26
2 РОЗРОБКА АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ АВТОМАТИЧНОГО УПРАВЛІННЯ ЕЛЕКТРОПРИЛАДАМИ	37
2.1 Огляд основних компонентів пристрою	37
2.2 Технічні особливості побудови системи та взаємодії компонентів	42
2.3 Налаштування безперебійного електроживлення головного контролера	48
3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ АВТОМАТИЧНОГО УПРАВЛІННЯ ЕЛЕКТРОПРИЛАДАМИ	52
3.1 Огляд основних програмних блоків та архітектури програми	52
3.2 Ключові функції прийому та передачі даних у системі	54
3.3 Базові принципи досягнення надійності програмного забезпечення	67
ВИСНОВКИ	70
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	73
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ	75

ВСТУП

Вибір теми кваліфікаційної роботи був зумовлений потребою у підвищенні зручності та ефективності використання побутових електроприладів, зниження витрат на електроенергію, а також необхідністю створення віртуальної системи, яку студенти університету можуть використовувати для вивчення засобів автоматизації та сучасних технологій Інтернету речей (IoT). Крім того, ця система може бути використана в реальних умовах для полегшення побутових задач.

Автоматизація управління побутовими електроприладами є актуальним завданням нашого часу. Люди навчилися використовувати різні технології для спрощення свого життя, але справжнім проривом стало впровадження IoT. Інтернет речей дозволяє об'єднати різноманітні пристрої в єдину мережу, забезпечуючи їх взаємодію та централізоване управління. Це дозволяє оптимізувати використання електроенергії, підвищити комфорт та безпеку житлових приміщень.

Провідна дослідницька компанія Transforma Insights випустила звіт по розвитку у галузі у світі, згідно з яким впровадження рішень IoT призведе до скорочення споживання електроенергії більш ніж на 1,6 петават-годин (hPh за рік) у 2028 році. Ця суттєва економія є результатом впровадження інтелектуальних систем моніторингу та управління енергією, які оптимізують споживання ресурсів, знижують втрати та підвищують загальну ефективність енергоспоживання.

Системи автоматичного управління електроприладами, засновані на IoT, дозволяють ефективно керувати освітленням, кліматичними умовами, системами безпеки та іншими аспектами розумного будинку за допомогою платформ, таких як Alexa, Google Home та SmartThings. Вони включають різноманітні датчики, контролери та виконавчі пристрої, які забезпечують автоматизацію рутинних задач. Наприклад, датчики освітленості та руху можуть автоматично регулювати освітлення залежно від наявності людей у приміщенні та рівня природного світла.

Інформаційною базою для написання кваліфікаційної роботи використовувалися електронні ресурси, публічно доступні матеріали досліджень та

аналітики в даному напрямку, спеціалізована та наукова література та інші джерела, серед яких звіт Transforma Insights, статті газети BIO Web Conf, науково-інформаційна соціальна мережа ResearchGate, тощо.

Об'єктом дослідження кваліфікаційної роботи є процес автоматизації управління побутовими електроприладами для підвищення зручності та ефективності використання електроенергії в домашніх умовах.

Предметом дослідження роботи є способи розробки та використання IoT-систем для автоматичного управління освітленням для догляду за кімнатними рослинами з використанням датчиків та мікроконтролера NodeMCU ESP8266.

Метою дослідження є дослідження сучасних технологій IoT та розробка концепції системи автоматичного управління освітленням для догляду за кімнатними рослинами, яка забезпечує зручність, енергоефективність та безпеку.

Відповідно до мети кваліфікаційної роботи було поставлено наступні задачі:

1. Дослідити сучасні технології IoT та їх застосування у системах розумного будинку.
2. Вивчити основні компоненти та протоколи передачі даних для систем автоматизації.
3. Ознайомитися з особливостями використання платформ автоматизації, таких як Alexa, Google Home та SmartThings.
4. На основі теоретичної підготовки розробити концепцію системи автоматичного управління освітленням для догляду за кімнатними рослинами на базі мікроконтролера NodeMCU ESP8266.
5. Реалізувати прототип системи та провести тестування її роботи в реальних умовах.

Методика дослідження складається з двох основних етапів – теоретичного і практичного. Спочатку вивчається проблема, проводиться пошук її рішень, пошук шляхів виконання завдань. Проводиться дослідження і теоретичне обґрунтування матеріалів та на основі отриманих даних розробляється план виконання практичної частини. Практичний етап включає в себе безпосередньо розробку

системи автоматичного управління – її апаратної частини та програмного забезпечення.

Наукова новизна дослідження полягає в тому, що для систем автоматизації догляду за кімнатними рослинами використовується мікроконтролер NodeMCU ESP8266 та різноманітні датчики, що дозволяє створювати ефективні та економічні рішення. Це дослідження також пропонує рекомендації щодо подальшого доопрацювання системи для підвищення її функціональності та надійності.

Практична значущість результатів дослідження полягає в можливості використовувати розроблену систему для покращення якості освітнього процесу в університетах та надання студентам можливості здобути нові практичні навички з розробки та роботи з подібними системами. Система також може використовуватися у реальних умовах для автоматизації догляду за кімнатними рослинами, забезпечуючи оптимальні умови для їх росту та розвитку.

1 ОГЛЯД ОСОБЛИВОСТЕЙ ФУНКЦІОНУВАННЯ СИСТЕМ РОЗУМНОГО БУДИНКУ

1.1 Сучасні системи управління електроприладами у розумному будинку

Сучасні технології проникають у всі сфери нашого життя, роблячи його більш зручним і ефективним. Одним з яскравих прикладів таких технологій є системи управління електроприладами у розумному будинку. Вони дозволяють інтегрувати різні електроприлади в єдину систему, що забезпечує їх централізоване управління і контроль. В рамках першого розділу кваліфікаційної роботи буде розглянуто три провідні системи управління розумним будинком: Amazon Alexa, Samsung Smart Things і Google Home, їх функціональні можливості, переваги та недоліки.

Amazon Alexa (рис. 1.1) є однією з найпопулярніших систем управління розумним будинком. Вона дозволяє користувачам керувати різними пристроями за допомогою голосових команд через інтегровані пристрої, такі як Echo, Echo Dot та інші. Alexa підтримує широкий спектр пристроїв, включаючи освітлення, термостати, камери безпеки та побутову техніку.



Рис 1.1 Вигляд апаратного забезпечення системи Amazon Alexa

Amazon Alexa знаходить застосування не лише в розумних будинках, але й у багатьох інших сферах. У сфері охорони здоров'я вона може використовуватися для нагадувань про прийом ліків та моніторингу стану пацієнтів. В освіті Alexa служить інструментом для інтерактивного навчання, допомагаючи студентам отримувати відповіді на запитання та виконувати завдання. У бізнесі IoT і Alexa забезпечують управління календарями, організацію нарад та автоматизацію офісних процесів. Вона може інтегруватися з системами управління завданнями та календарями, допомагаючи співробітникам залишатися організованими і продуктивними. Alexa може автоматично нагадувати про зустрічі, створювати звіти та навіть керувати системами офісного освітлення та клімат-контролю. Це дозволяє бізнесам підвищувати ефективність і знижувати операційні витрати. В автомобільній індустрії IoT і Alexa інтегруються в автомобілі, дозволяючи водіям керувати навігацією, музикою та іншими функціями за допомогою голосу, що підвищує безпеку та комфорт на дорозі.

Amazon Alexa має ряд переваг та недоліків, які тезисно показані на рис. 1.2.

Переваги Amazon Alexa	Недоліки Amazon Alexa
<p>1. Широка підтримка пристроїв: Alexa сумісна з багатьма виробниками, що дозволяє інтегрувати різноманітні електроприлади в єдину систему.</p> <p>2. Голосове управління: Користувачі можуть виконувати різні дії, просто говорячи команди, що робить систему дуже зручною.</p> <p>3. Інтеграція з іншими сервісами: Alexa може взаємодіяти з багатьма популярними сервісами, такими як Spotify, Amazon Prime, Uber та іншими.</p>	<p>1. Конфіденційність даних: Існують побоювання щодо збору і використання даних користувачів компанією Amazon.</p> <p>2. Залежність від інтернету: Для роботи системи необхідне стабільне інтернет-з'єднання.</p>

Рис 1.2 Переваги та недоліки системи Amazon Alexa

Архітектура системи Amazon Alexa складається з декількох ключових компонентів, які працюють разом для забезпечення ефективного управління

розумним будинком. Ці компоненти включають пристрої користувачів, Alexa Voice Service (AVS), хмарну інфраструктуру Amazon Web Services (AWS) та інтегровані розумні пристрої (рис. 1.3).

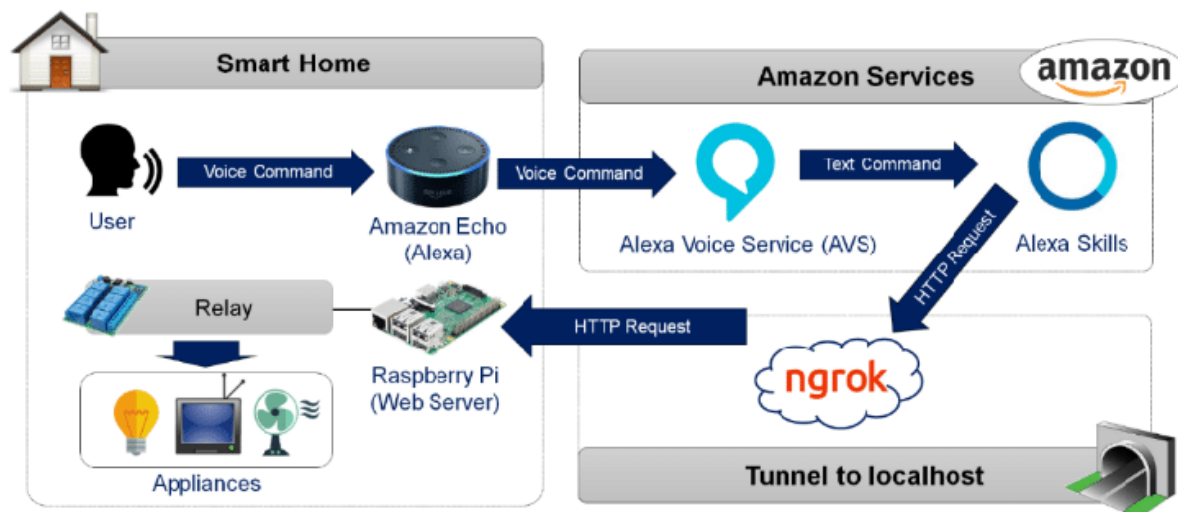


Рис 1.3 Структурна схема взаємодії Amazon Services та розумного будинку [1]

Основні пристрої, які взаємодіють з системою Amazon Alexa, включають Echo пристрої, такі як Echo, Echo Dot, і мобільні додатки, доступні на iOS і Android. Echo пристрої є основними апаратними пристроями, через які користувачі взаємодіють з Alexa. Вони оснащені мікрофонами, динаміками та підтримують бездротове підключення до інтернету.

Мобільні додатки дозволяють користувачам налаштовувати і керувати своїми пристроями, створювати сценарії автоматизації і отримувати доступ до різних сервісів.

Alexa Voice Service (AVS) є основним компонентом, що забезпечує обробку голосових команд і відповіді на них. AVS складається з таких підкомпонентів: розпізнавання мови (ASR), розуміння природної мови (NLP) та синтезу мови (TTS). ASR перетворює голосові команди у текст, NLP інтерпретує сенс команд і визначає відповідні дії, а TTS дозволяє AVS відповідати користувачам голосом.

Хмарна інфраструктура Amazon Web Services (AWS) надає масштабовану і надійну інфраструктуру для обробки, зберігання і передачі даних, необхідних для

роботи Alexa. Основні сервіси AWS, що використовуються, включають Lambda, DynamoDB, S3 (Simple Storage Service) та IAM (Identity and Access Management). Lambda виконує код у відповідь на тригери, обробляючи голосові команди і взаємодіючи з іншими сервісами. DynamoDB зберігає інформацію про користувачів, пристрої і налаштування, S3 використовується для зберігання медіафайлів та інших даних, а IAM керує доступом і безпекою.

Amazon Alexa підтримує широкий спектр розумних пристроїв від різних виробників, включаючи освітлення, термостати, камери безпеки, замки та інше. Інтеграція здійснюється через стандартні протоколи, такі як Zigbee, Z-Wave та Wi-Fi, а також через API (Application Programming Interface).

Пристрої Echo, такі як Amazon Echo, створені для того, щоб полегшити взаємодію з цифровим світом за допомогою голосових команд. Коли користувач дає голосову команду, наприклад, "Алекса, ввімкни світло", пристрій Echo захоплює аудіосигнал і передає його на сервери Amazon Voice Services (AVS). Цей процес включає кілька етапів обробки.

Розробники також мають можливість розширювати функціональні можливості Alexa за допомогою Alexa Skills Kit (ASK). Цей інструмент дозволяє створювати нові "навички" (skills) для Alexa, забезпечуючи користувачів новими можливостями та функціями. Навички можуть бути орієнтовані на різні сфери: від контролю розумного будинку до інтерактивних ігор і навчання. Еталонна архітектура Alexa Skills Kit показана на рис. 1.4.

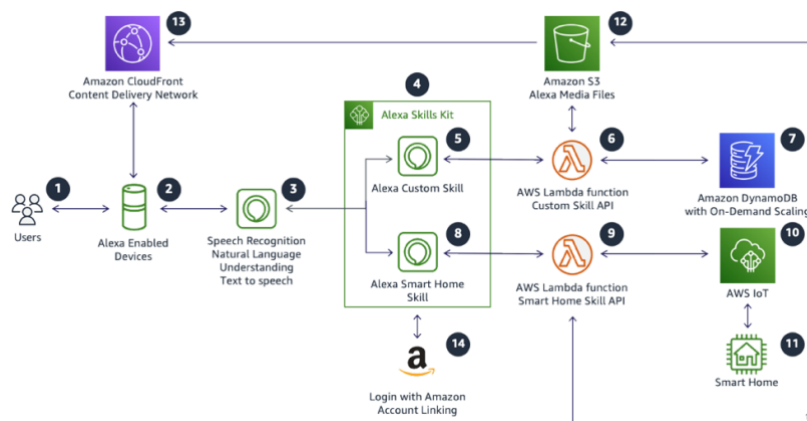


Рис 1.4 Еталонна архітектура Alexa Skills Kit

Alexa використовує різні методи для управління розумними пристроями: деякі пристрої підключаються безпосередньо до Echo через Zigbee або інші протоколи, інші взаємодіють через API, що дозволяє Alexa надсилати команди до хмарних сервісів виробників пристроїв. Користувачі можуть створювати сценарії автоматизації за допомогою додатку Alexa. Це дозволяє налаштувати послідовність дій, які виконуються автоматично за умовами часу, місця або інших тригерів.

Архітектура системи Amazon Alexa являє собою складний набір компонентів, які працюють разом для забезпечення ефективного і зручного управління розумним будинком. Інтеграція голосових технологій, хмарних сервісів і розумних пристроїв дозволяє користувачам легко контролювати свої будинки, роблячи їх більш комфортними і функціональними. Amazon продовжує розвивати Alexa, додаючи нові можливості та вдосконалюючи існуючі функції, що робить систему однією з провідних на ринку розумних будинків.

Samsung SmartThings є ще однією популярною системою для управління енергопристроями, яка забезпечує інтеграцію різноманітних розумних пристроїв у єдину екосистему, що сприяє зручності, безпеці та енергоефективності. Архітектура цієї системи базується на поєднанні апаратного та програмного забезпечення, що забезпечує взаємодію та координацію між різними компонентами розумного будинку.

Ця платформа дозволяє користувачам об'єднувати та керувати розумними пристроями, такими як світильники, термостати, замки, датчики руху, розетки та інші, через єдиний інтерфейс. За допомогою мобільного додатка або голосових асистентів, користувачі можуть налаштовувати сценарії автоматизації, які оптимізують використання енергії, підвищують рівень комфорту та забезпечують безпеку в будинку.

Samsung SmartThings має ряд переваг та недоліків, які тезисно показані на рис.

1.5



Рис 1.5 Переваги та недоліки системи Samsung SmartThings

Основою системи Samsung SmartThings є центральний хаб (SmartThings Hub)(рис.1.6), який слугує комунікаційним центром для всіх підключених пристроїв. Хаб підтримує різні бездротові протоколи, такі як Zigbee, Z-Wave та Wi-Fi, що забезпечує широкі можливості для підключення різноманітних розумних пристроїв від різних виробників.



Рис 1.6 Контролер розумного будинку SmartThings

Ключовим елементом архітектури є мобільний додаток SmartThings, який дозволяє користувачам керувати своїми розумними пристроями, створювати сценарії автоматизації та отримувати сповіщення про стан системи.

Додаток забезпечує інтуїтивно зрозумілий інтерфейс для налаштування та моніторингу пристроїв, а також дозволяє інтегруватися з голосовими асистентами, такими як Samsung Vixby, Google Assistant та Amazon Alexa, що додає додатковий рівень зручності через голосове керування.

Архітектура системи передбачає використання хмарних сервісів Samsung для забезпечення віддаленого доступу та зберігання даних. Хмарна платформа SmartThings Cloud забезпечує синхронізацію між хабом та мобільним додатком, пристроями (рис. 1.7), а також дозволяє користувачам керувати своїм розумним будинком з будь-якої точки світу. Хмарні сервіси також відіграють ключову роль у забезпеченні безпеки даних, використовуючи сучасні методи шифрування та аутентифікації для захисту інформації користувачів.

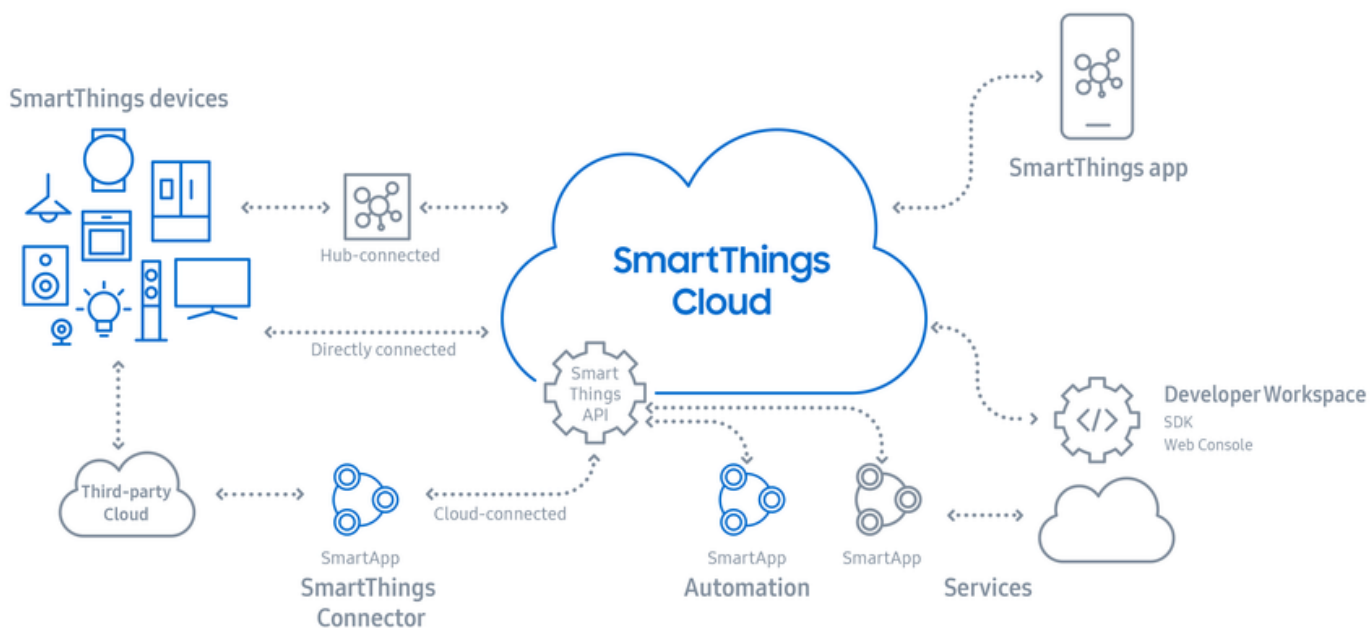


Рис 1.7 Структурна схема взаємодії елементів системи через хмарну платформу SmartThing Cloud

Хмарна платформа SmartThings Cloud синхронізує всі підключені пристрої, дозволяючи їм взаємодіяти між собою. Наприклад, дані з датчика руху можуть автоматично передаватися на розумні світильники, які вмикаються при виявленні

руху. Це створює інтелектуальну та адаптивну екосистему, що реагує на потреби користувачів і забезпечує максимальний комфорт.

Важливою складовою архітектури Samsung SmartThings є підтримка відкритих API та SDK, що дозволяє розробникам створювати власні додатки та інтеграції для розширення функціональності системи. Це сприяє зростанню екосистеми SmartThings та забезпечує сумісність з широким спектром розумних пристроїв, підвищуючи гнучкість та адаптивність системи до потреб користувачів

Огляд архітектури Home API та того, як вона інтегрується з SmartThings та стороннім додатком, можна побачити на рис. 1.8.

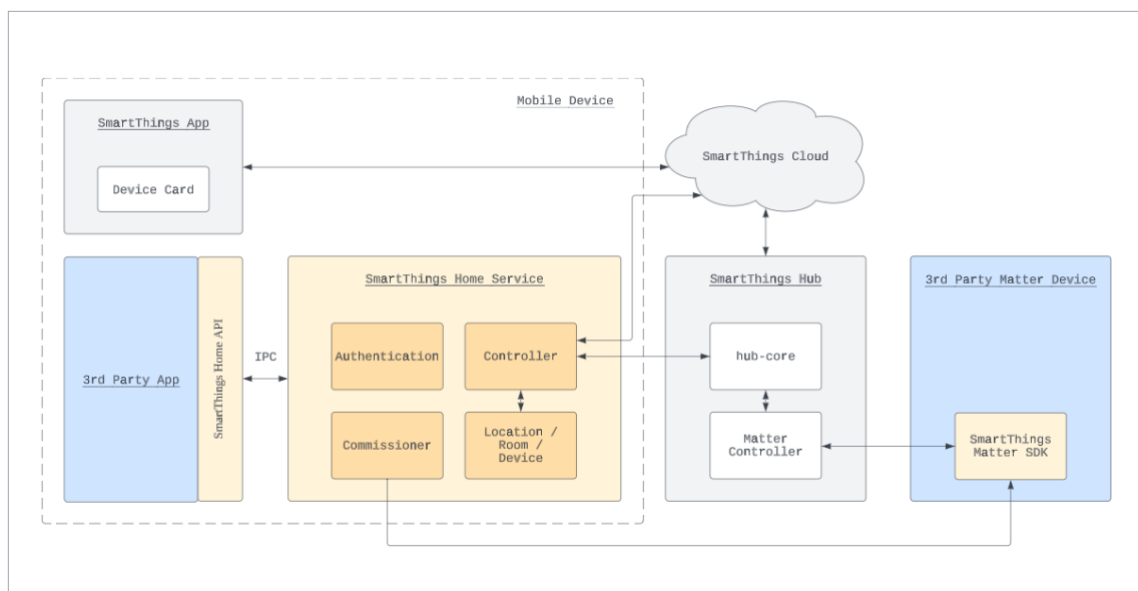


Рис 1.8 Інтеграція Home API з SmartThings та додатком

Одним з головних аспектів архітектури є надійність та масштабованість системи. Завдяки модульній структурі, SmartThings дозволяє користувачам поступово розширювати свою систему розумного будинку, додаючи нові пристрої та функції без необхідності суттєвих змін у вже існуючій інфраструктурі.

У підсумку, архітектура системи розумного будинку Samsung SmartThings базується на поєднанні центрального хабу, мобільного додатку та хмарних сервісів, що забезпечують інтеграцію, керування та автоматизацію різноманітних розумних

пристроїв. Відкрита платформа та підтримка різних бездротових протоколів роблять SmartThings гнучкою та масштабованою системою, здатною адаптуватися до індивідуальних потреб користувачів та забезпечувати високий рівень комфорту, безпеки та енергоефективності в сучасних розумних будинках.

Ще однією системою управління розумним будинком є Google Home Google Home, що представляє собою платформу для управління розумним будинком, розроблену компанією Google. Вона об'єднує різноманітні пристрої та сервіси, забезпечуючи їх інтеграцію в єдину екосистему. Архітектура Google Home дозволяє користувачам контролювати свої розумні пристрої за допомогою голосових команд через Google Assistant, а також за допомогою мобільного додатку Google Home.

Google Assistant(рис. 1.9) виступає центральним елементом системи, виконуючи роль інтерфейсу між користувачем і розумними пристроями. Завдяки голосовим командам користувачі можуть легко контролювати та налаштовувати свої пристрої, забезпечуючи зручність і інтуїтивну зрозумілість взаємодії з системою. Додаток Google Home є ще одним ключовим компонентом, що слугує інструментом для налаштування, управління та моніторингу всіх підключених пристроїв. Через додаток користувачі можуть додавати нові пристрої, створювати групи пристроїв, налаштовувати автоматизацію та отримувати сповіщення про стан системи. Це забезпечує централізоване управління, яке дозволяє зручно контролювати всі аспекти розумного будинку з одного місця.



Рис 1.9 Google Assistant

Google Home використовує модель інтеграції "Cloud-to-Cloud" (C2C) для забезпечення взаємодії з різноманітними смарт-пристроями та сервісами, розташованими в різних хмарних екосистемах. Ця модель дозволяє розширити функціональність Google Home, забезпечуючи інтеграцію з широким спектром пристроїв від різних виробників, що використовують власні хмарні платформи.

Переваги Cloud-to-Cloud інтеграцій включають широкую сумісність, покращену функціональність та оновлення в реальному часі. Завдяки C2C інтеграціям, Google Home може працювати з величезною кількістю пристроїв від різних виробників, не залежачи від апаратних обмежень.

Інтеграція різних хмарних сервісів дозволяє створювати більш комплексні та інтелектуальні сценарії автоматизації, які враховують дані з різних джерел. Двостороння комунікація та автоматичні оновлення стану забезпечують актуальність інформації про підключені пристрої, що покращує користувацький досвід.

Технологія Cloud-to-Cloud відноситься до безшовної інтеграції та взаємодії між різними хмарними сервісами і платформами, що дозволяє передавати, синхронізувати та автоматизувати дані між кількома хмарними середовищами. Це дозволяє бізнесам використовувати переваги різних постачальників хмарних послуг та будувати більш гнучкі, масштабовані та ефективні ІТ-інфраструктури

Cloud-to-Cloud інтеграція відкриває нові можливості для об'єднання ресурсів з різних хмарних платформ(рис. 1.10) Компанії можуть використовувати Amazon Web Services (AWS), Google Cloud Platform (GCP) для аналітики даних і Microsoft Azure для розгортання корпоративних додатків. Така гнучкість забезпечує більш ефективне управління ресурсами та зниження витрат.

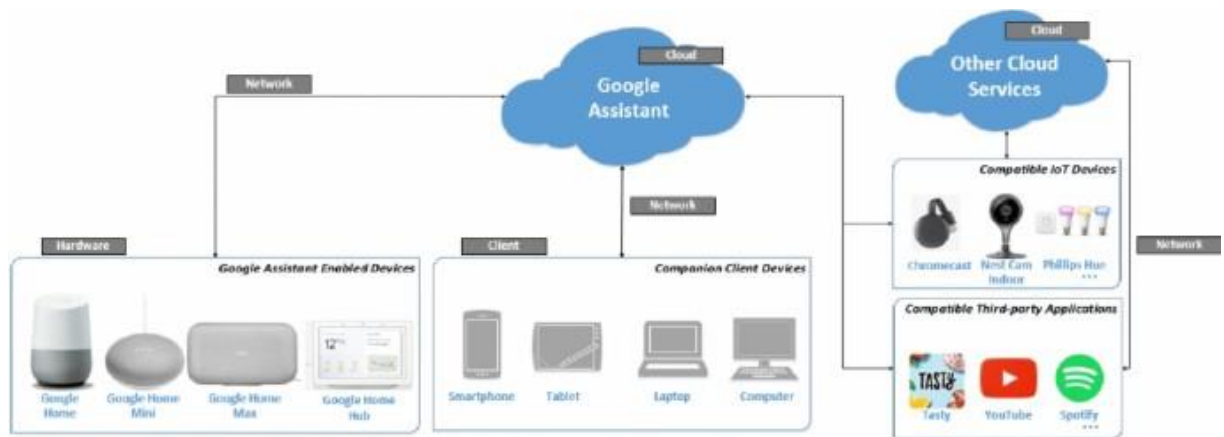


Рис 1.10 Приклад взаємодії між різними хмарними сервісами за допомогою Cloud-to-Cloud

Одним з ключових елементів архітектури Google Home, що забезпечує ефективність та інтеграцію розумного будинку, є Home Graph (рис.1.11). Це технологія, розроблена компанією Google, яка зберігає та обробляє інформацію про розумні пристрої, їхній статус та взаємозв'язки між ними. Home Graph виконує роль центральної бази даних для всіх пристроїв, яка є логічною картою дому, яка дозволяє вести природну розмову з Google Assistant,

Home Graph зберігає інформацію про всі підключені пристрої, включаючи їхні характеристики, поточний стан та розташування в будинку. Завдяки цьому Google Assistant має можливість точно інтерпретувати команди користувачів та виконувати їх у відповідності до поточного контексту. Наприклад, коли користувач дає команду "Вимкни світло в спальні", Home Graph допомагає Google Assistant визначити, які саме лампочки розташовані в спальні, та коректно виконати команду.

Також, Home Graph підтримує складні сценарії автоматизації, враховуючи взаємозв'язки між пристроями. Це дозволяє створювати багатоступеневі сценарії, які реагують на різноманітні умови та події.

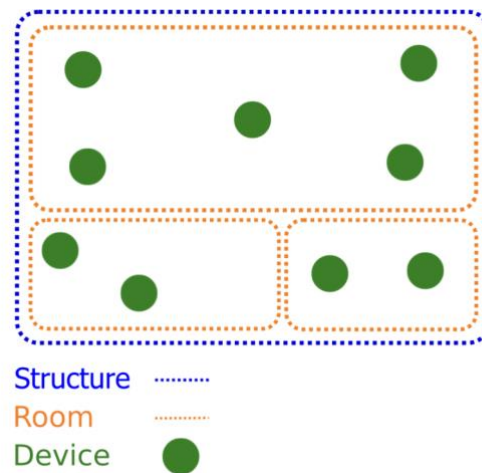


Рис 1.11 Приклад логічної карти будинку Home Graph

Завдяки використанню Home Graph, Google Home надає користувачам можливість управляти розумним будинком більш інтелектуально та контекстуально обізнано. Ця технологія забезпечує високу ефективність, безпеку та гнучкість системи, дозволяючи інтегрувати різноманітні пристрої та сервіси в єдину екосистему. Таким чином, Home Graph є важливим елементом архітектури Google Home, який сприяє створенню сучасного та комфортного життєвого простору, що відповідає найвищим стандартам безпеки та зручності.

У підсумку, архітектура Google Home є потужною та гнучкою платформою для управління розумним будинком. Вона забезпечує інтеграцію різноманітних пристроїв, використовуючи сучасні протоколи зв'язку та надійні механізми безпеки. Завдяки централізованому управлінню через додаток Google Home та можливостям автоматизації, користувачі можуть легко контролювати та оптимізувати роботу свого розумного будинку.

1.2 Дослідження ефективності автоматичного управління електроприладами для оптимізації використання електроенергії

У сучасному світі, де зростаючі потреби в енергії зустрічаються з обмеженими ресурсами та необхідністю зменшення викидів парникових газів,

ефективне використання електроенергії стає надзвичайно важливим завданням. Одним із перспективних підходів до вирішення цього питання є впровадження автоматизованих систем управління електроприладами на основі технологій Інтернету речей (IoT).

Автоматизація управління електроприладами за допомогою IoT базується на інтеграції різних сенсорів, контролерів і програмного забезпечення, що дозволяє в режимі реального часу отримувати дані про споживання електроенергії, аналізувати їх та автоматично регулювати роботу приладів. Це, в свою чергу, сприяє підвищенню енергоефективності, зменшенню навантаження на електромережу та зниженню загальних витрат на електроенергію.

Провідна дослідницька компанія Transforma Insights випустила звіт по розвитку у галузі у світі, згідно з яким впровадження рішень IoT призведе до скорочення споживання електроенергії більш ніж на 1,6 петаватт-годин (hPh за рік) у 2028 році (рис 1.12). Ця суттєва економія є результатом впровадження інтелектуальних систем моніторингу та управління енергією, які оптимізують споживання ресурсів, знижують втрати та підвищують загальну ефективність енергоспоживання.

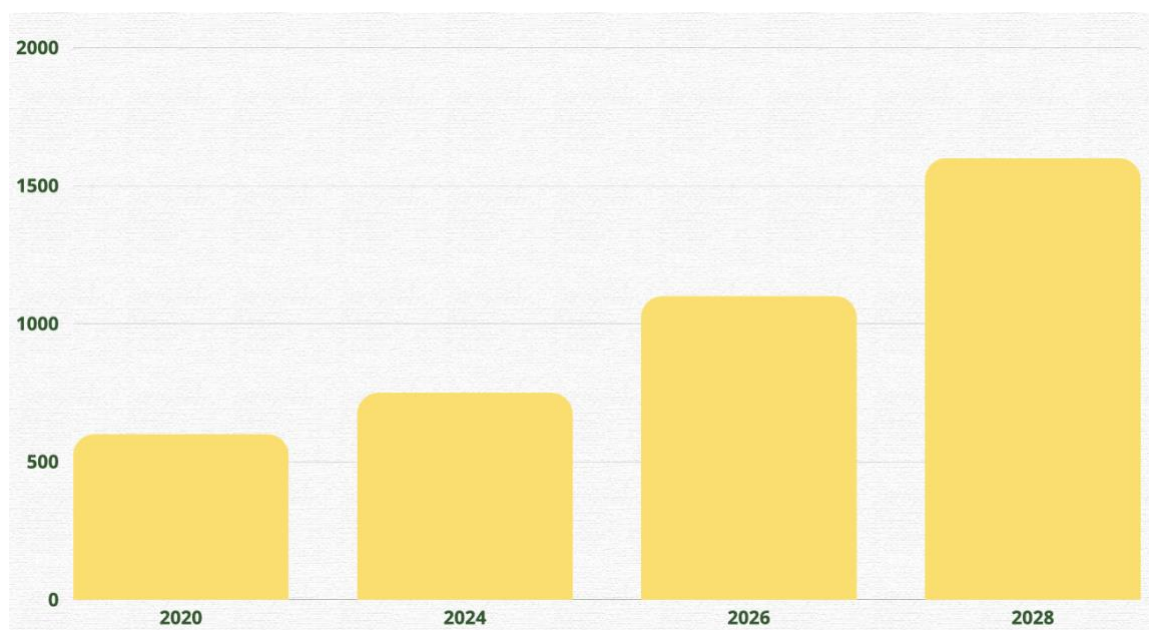


Рис 1.12 Прогноз економії електроенергії за допомогою впровадження ІОТ рішень в hPh за рік

Особливу увагу варто звернути на використання ІОТ для розумних будинків, які є одним із найперспективніших напрямків впровадження ІоТ технологій для підвищення енергоефективності та зручності.

Автоматизація систем опалення, вентиляції та кондиціонування повітря (HVAC) є одним з найбільших споживачів енергії в житлових будинках. Використання розумних термостатів, які можуть автоматично регулювати температуру на основі присутності людей у приміщенні та зовнішніх погодних умов, дозволяє значно знизити енергоспоживання.

Дослідження показують, що такі системи можуть знизити споживання енергії на 10-15%, що сприяє зменшенню витрат на електроенергію та підвищенню комфортності проживання.

Іншим важливим аспектом є використання розумних освітлювальних систем. Завдяки ІоТ, освітлення може бути автоматично налаштоване відповідно до присутності людей у приміщенні, часу доби та рівня природного освітлення. Це дозволяє знизити споживання електроенергії на освітлення до 20%, що особливо важливо в великих житлових комплексах та офісних будівлях.

В статті газети BIO Web Conf. провели ретельний аналіз витрат і вигод для об'єктивної оцінки економічної життєздатності інвестицій у розумні будинки. енергоефективності та зниження експлуатаційних витрат, а також початкові інвестиційні витрати на придбання та встановлення обладнання розумного дому.

Для реалізації проекту з автоматизації та підвищення енергоефективності було закуплено обладнання для розумного дому на суму 3750 доларів. Ця інвестиція включає в себе широкий спектр розумних пристроїв, які спрямовані на зниження споживання електроенергії та оптимізацію побутових процесів.

Розподілення бюджету по обладнанню було зображено на рис. 1.13

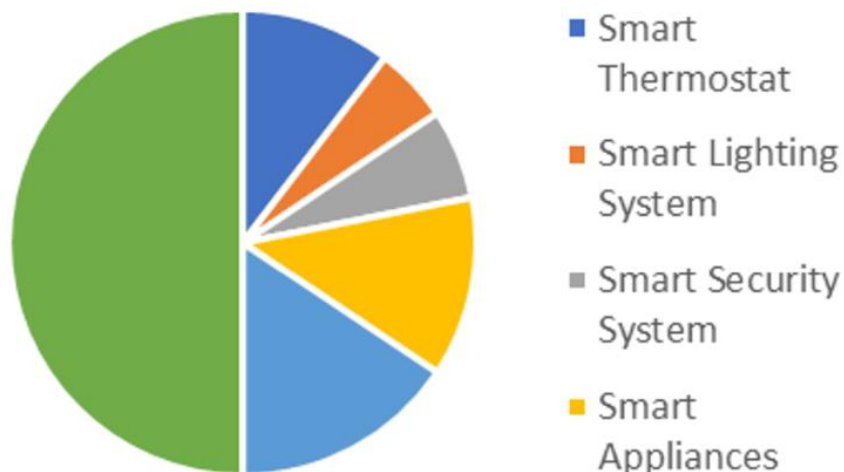


Рис 1.13 Розподілення витрат на обладнання ІОТ

Учасники проекту з автоматизації розумного будинку, який передбачав впровадження технологій Інтернету речей (ІоТ), змогли досягти значних результатів у зниженні споживання електроенергії. Встановлення розумних термостатів, систем розумного освітлення, розумних побутових приладів та систем моніторингу енергоспоживання дозволило суттєво зменшити витрати на електроенергію. Щомісяця учасники економили в середньому 240 кВт/год електроенергії за рахунок (табл. 1.1)

Таблиця 1.1

Щомісячне заощадження електроенергії (кWh) за рахунок впровадження ІОТ пристроїв

Розумний пристрій	Щомісячне заощадження електроенергії (кWh)
Розумний термостат	50
Розумна система освітлення	25
Розумна система захисту	30
Розумна побутова техніка	60
Розумні системи HVAC (опалення, вентиляції та кондиціонування повітря)	75
Загальні місячні заощадження	240

Ці результати демонструють значний потенціал технологій IoT у забезпеченні енергоефективності та зниженні витрат на електроенергію. Впровадження таких технологій у масштабах, як окремих домогосподарств, так і великих будівель, може суттєво знизити загальне енергоспоживання, що сприятиме економії ресурсів та покращенню екологічної ситуації, що дозволило учасникам проекту суттєво знизити споживання електроенергії та заощадити значні кошти. В таблиці 1.2 наведено дані про щомісячне заощадження електроенергії.

Таблиця 1.2

Річне заощадження коштів за рахунок впровадження IoT пристроїв

Категорія	Щорічна економія (USD)
Економія на енергії	1800
Економія на обслуговуванні	600
Економія на підписках	700
Загальна щорічна економія	3120

Таким чином, впровадження технологій Інтернету речей (IoT) та автоматизації в розумний будинок показало значні технічні та фінансові результати. Використання розумних термостатів, систем розумного освітлення, розумної побутової техніки та систем моніторингу енергоспоживання продемонструвало, що ці технології можуть суттєво зменшити споживання електроенергії та покращити енергоефективність.

1.3 Переваги та недоліки застосування популярних протоколів передачі даних у розумних будинках

Розумні будинки, інтегровані з численними пристроями Інтернету речей (IoT), залежать від ефективних та надійних протоколів передачі даних для забезпечення безперервного функціонування систем. Безпроводні та дротові

протоколи передачі даних забезпечують комунікацію між різними компонентами розумного будинку, такими як датчики, камери, термостати та освітлювальні системи. Ці протоколи дозволяють пристроям обмінюватися інформацією, координувати свої дії та автоматизувати процеси для підвищення комфорту, безпеки та енергоефективності.

Серед популярних протоколів передачі даних, які використовуються в розумних будинках, виділяються безпроводні та дротові технології. Безпроводні протоколи забезпечують гнучкість у розміщенні пристроїв та спрощують процес їхнього підключення та налаштування. Вони дозволяють створювати розумні будинки без необхідності прокладати додаткові кабелі, що знижує витрати на встановлення та підтримку системи.

На рисунку 1.14 зображено використання безпроводних протоколів у розумних будинках

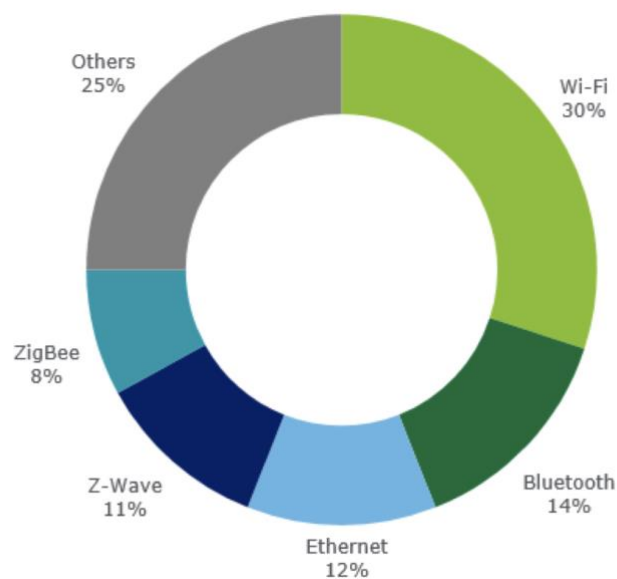


Рис 1.14 Використання безпроводних протоколів в системах розумного дому

Wi-Fi є одним з найпоширеніших протоколів передачі даних у розумних будинках завдяки своїй високій швидкості та стабільності з'єднання.

Технологія WiFi призначена для підключення електронних пристроїв у бездротовій локальній мережі (WLAN). WiFi базується на сімействі стандартів

IEEE 802.11, які працюють у неліцензованих діапазонах 2,4 ГГц та 5 ГГц, доступних у всьому світі. Стандарти IEEE 802.11b/g/n використовують діапазон 2,4 ГГц, тоді як IEEE 802.11a/n/ac працюють на частоті 5 ГГц. Використовуючи 14 частково перекриваючих смуг шириною 22 МГц у діапазоні 2,4 ГГц, WiFi має величезну пропускну здатність і, як результат, дозволяє досягати дуже високих швидкостей передачі даних. Лише 13 із цих каналів доступні в Європі, 11 – у США, і лише один – у Японії. Швидкість передачі даних становить 54 Мбіт/с. Основні переваги та недоліки WiFi протоколу зображено на рисунку 1.15

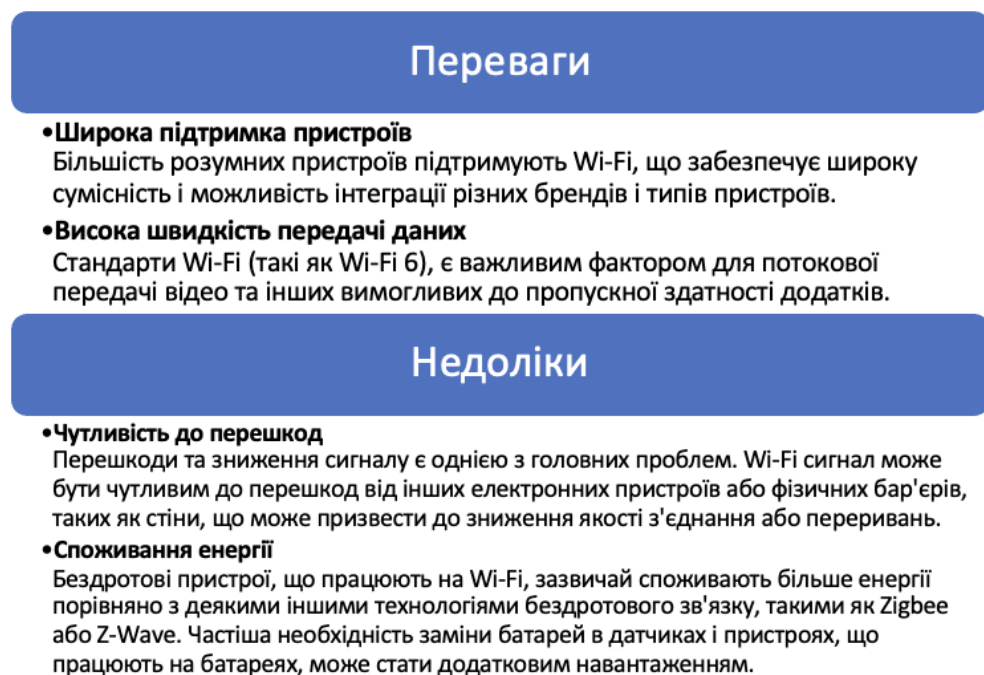


Рис 1.15 Основні переваги та недоліки WiFi протоколу в системі розумного будинку

Оскільки технологія WiFi домінує в сегменті споживчої електроніки, завжди краще використовувати існуючу WiFi інфраструктуру для нових IoT застосувань, особливо якщо їм потрібно спілкуватися зі смартфонами, ноутбуками, планшетами, телевізорами, ігровими консолями тощо. Крім того, WiFi має вбудовану сумісність з IP. Це означає, що немає необхідності купувати дорогі шлюзи для підключення IoT пристроїв до Інтернету.

Wi-Fi є зручним і гнучким рішенням для автоматизації розумного дому завдяки своїй простоті установки, високій швидкості передачі даних та широкій підтримці пристроїв. Однак, для забезпечення стабільної роботи і безпеки, необхідно враховувати можливі перешкоди, вразливості безпеки та енергоспоживання. Комбінація Wi-Fi з іншими технологіями (наприклад, Zigbee або Z-Wave) може забезпечити оптимальне рішення для створення надійної та ефективної мережі розумного дому.

ZigBee є стандартом для низькопотужного, низькошвидкісного бездротового зв'язку, який спрямований на забезпечення сумісності та охоплює пристрої від різних виробників. Протокольний стек, наданий ZigBee, є відкритим вихідним кодом і безкоштовним для використання будь-яким розробником або компанією. Zigbee є єдиним глобальним бездротовим рішенням на основі стандартів, яке може зручно та економічно керувати найширшим спектром пристроїв для покращення комфорту, безпеки та зручності для споживачів.

ZigBee базується на фізичному рівні та рівні управління доступом до середовища, визначених у стандарті IEEE 802.15.4. Він використовує три неліцензовані частотні діапазони залежно від місця розташування: від 2400 МГц до 2483,5 МГц, від 902 МГц до 928 МГц і від 868 МГц до 868,6 МГц. У діапазоні 2,4 ГГц виділено 16 каналів по 5 МГц кожний, у діапазоні 915 МГц – 10 каналів, а в діапазоні 868 МГц – лише один канал. У діапазонах 868 і 915 МГц використовується двійкова фазова маніпуляція (BPSK), а в діапазоні 2,4 ГГц – офсетна квадратична фазова маніпуляція (OQPSK). Пряме спектральне розширення (DSSS) допомагає підвищити стійкість до сильних перешкод. Типовий діапазон дії ZigBee складає близько 10-100 м, залежно від потужності та характеристик середовища. На рисунку 1.16 зображено технічні характеристики ZigBee та інших протоколів.

	Wi-Fi 802.11n	Bluetooth	Bluetooth LE	ZigBee	Z-Wave
Frequency	2.4–5.8 GHz	2.402–2.48 GHz	2.402–2.48 GHz	868/915 MHz, 2.4 GHz	868/915 MHz
Data rate	450 Mbps	0.7–2.1 Mbps	2 Mbps	20/40 kbps, 250 kbps	10–100 kbps
Range	10–100 m	15–20 m	10–15 m	10–100 m	30–50 m
Network size	Thousands (mesh)	8	N/A	65,536	232
Network Topology	Star, tree, P2P, mesh	Star	Star	Star, mesh, cluster tree	Mesh
Encryption	WPA2	AES-128	AES-128	AES-128	AES-128

Рис 1.16 Технічні характеристики ZigBee в порівнянні з другими протоколами

ZigBee має кілька значних переваг, які роблять його привабливим для використання в системах розумного будинку та інших додатках Інтернету речей (IoT). По-перше, низьке енергоспоживання є однією з ключових переваг ZigBee. Ця технологія спеціально розроблена для пристроїв, що працюють від батарей, забезпечуючи тривалий термін їхньої роботи без необхідності частої заміни або підзарядки. Це особливо важливо для датчиків та інших пристроїв, розташованих у важкодоступних місцях.

Важливою перевагою є стійкість до перешкод. Використання таких технологій, як пряме спектральне розширення (DSSS) та різні методи модуляції, підвищує стійкість ZigBee до перешкод і забезпечує надійний зв'язок навіть у складних умовах. Це робить ZigBee надійним вибором для використання в насичених середовищах з великою кількістю бездротових пристроїв.

Ще одна перевага стосується діапазону дії. ZigBee забезпечує типовий діапазон дії від 10 до 100 метрів, що є достатнім для більшості додатків у розумних будинках. Крім того, завдяки сітковій топології мережі (mesh network), ZigBee може охоплювати більші площі за рахунок ретрансляції сигналу через інші пристрої.

Однак, поряд з численними перевагами, ZigBee має й певні недоліки. Одним з основних недоліків є низька пропускна здатність порівняно з іншими протоколами, такими як WiFi. Це обмежує використання ZigBee для передачі великих обсягів даних, що може бути критичним для деяких застосувань, наприклад, для відеоспостереження.

Ще одним значним недоліком є відсутність зворотної сумісності у профілі ZigBee IP. Новий профіль ZigBee IP забезпечує IP-зв'язок для пристроїв ZigBee, але не підтримує зворотну сумісність з попередніми версіями, що може створювати проблеми при інтеграції зі старими пристроями.

На відміну від ZigBee, інший популярний протокол Z-Wave є світовим лідером на ринку бездротового управління з понад 70 мільйонами проданих продуктів по всьому світу, підтримуваних більш ніж 450 виробниками. Основною метою Z-Wave є забезпечення надійної передачі коротких повідомлень від контрольного пристрою до інших вузлів у мережі.

Мережа Z-Wave(рис.1.17) складається з одного шлюзу, одного контролера і щонайменше одного керованого пристрою. Існують два типи пристроїв: контролери та підлеглі. Контролери надсилають команди підлеглим, які відповідають на запити або виконують команди. Підлеглі можуть діяти як маршрутизатори в часокритичних і подієвих додатках, таких як сигналізації. Максимальний розмір мережі обмежений до 232 вузлів. Фізичний та контрольний рівні доступу до середовища визначені в Рекомендації ITU-T G.9959. Максимальний діапазон становить 30 метрів. Залежно від регіону, Z-Wave використовує дещо різні частоти: 868,42 МГц для Європи, 908,4 МГц для США, 919,8 МГц для Австралії та Бразилії тощо. Оскільки він працює в діапазоні нижче 1 ГГц, він не чутливий до перешкод від WiFi та інших бездротових технологій у діапазоні 2,4 ГГц.

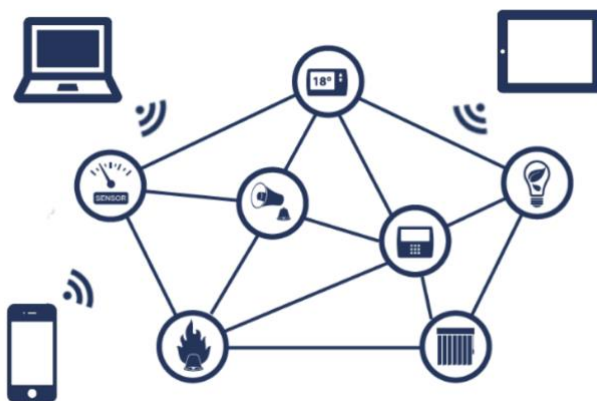


Рис 1.17 Топологія мережі Z-Wave

Z-Wave має кілька значних переваг, які роблять його привабливим вибором для використання в системах розумного будинку та інших додатках Інтернету речей (IoT). Однією з основних переваг Z-Wave є його висока надійність передачі даних. Протокол спеціально розроблений для забезпечення надійної передачі коротких повідомлень від контрольного пристрою до інших вузлів у мережі, що особливо важливо в критичних додатках, таких як системи безпеки та сигналізації.

Іншою важливою перевагою Z-Wave є його низька чутливість до перешкод. Оскільки Z-Wave працює в діапазоні нижче 1 ГГц, він не чутливий до перешкод від WiFi та інших бездротових технологій, які працюють у діапазоні 2,4 ГГц. Це забезпечує стабільну роботу пристроїв навіть у насичених бездротових середовищах. Крім того, використання модуляції частотного ключування (Gaussian frequency-shift keying) забезпечує високу стійкість до перешкод і підвищує надійність передачі даних.

Протокол Z-Wave має ряд недоліків, які можуть вплинути на його ефективність та доцільність використання у системах автоматизації розумного дому.

Одним з основних недоліків Z-Wave є висока вартість пристроїв, що використовують цей протокол. Порівняно з іншими технологіями автоматизації, такими як Wi-Fi або Bluetooth, пристрої на базі Z-Wave часто мають вищу ціну. Це пов'язано з тим, що Z-Wave спеціально розроблений для автоматизації, що забезпечує високу надійність і безпеку, але водночас підвищує витрати на виробництво та впровадження таких пристроїв. Відповідно, вартість розумних замків, термостатів або датчиків на основі Z-Wave може бути значно вищою, ніж аналогічних пристроїв з іншими протоколами, що може обмежити його застосування в деяких випадках.

Частотний діапазон, що використовується Z-Wave, також є недоліком. Z-Wave використовує різні частотні діапазони в різних регіонах світу, що може ускладнити сумісність пристроїв, придбаних в різних країнах. Наприклад, пристрої, придбані в США, можуть не працювати в Європі без належного

налаштування, що може створити проблеми для користувачів, які подорожують або переїжджають. Це також ускладнює міжнародний ринок пристроїв на базі Z-Wave, вимагаючи від виробників адаптувати свої продукти до різних регіональних стандартів.

Також, одним з популярних бездротовим протоколів є Bluetooth, який широко використовується для підключення різних пристроїв, включаючи пристрої автоматизації у "розумних будинках". Як і будь-яка технологія, Bluetooth має свої переваги та недоліки(рис.1.18)

Переваги	Недоліки
<ul style="list-style-type: none"> •Низьке енергоспоживання Сучасні версії Bluetooth, такі як Bluetooth Low Energy (BLE), споживають дуже мало енергії, що робить їх ідеальними для пристроїв, що працюють на батареях. Це особливо корисно для сенсорів і маленьких гаджетів у "розумних будинках", які потребують тривалого часу роботи без частих заміन батарей. •Автоматичне сполучення Функція автоматичного сполучення полегшує підключення пристроїв без необхідності введення складних налаштувань. Це зручно для користувачів, які можуть швидко налаштувати свої пристрої. 	<ul style="list-style-type: none"> •Обмежений радіус дії Одним з основних недоліків Bluetooth є обмежений радіус дії, який зазвичай становить від 10 до 100 метрів в залежності від версії і класу пристрою. Це може бути недостатнім для великих будинків або багатоповерхових будівель, де пристрої знаходяться далеко один від одного. •Перешкоди і затримки Bluetooth сигнал може піддаватися впливу перешкод від інших бездротових пристроїв або фізичних бар'єрів, таких як стіни і меблі, що може призводити до зниження якості з'єднання, затримок або втрати даних. Це може негативно вплинути на стабільність і надійність зв'язку.

Рис 1.18 Переваги та недоліки Bluetooth протоколу

Технологія Bluetooth базується на стандарті IEEE 802.15.1 для бездротового зв'язку на короткі відстані між фіксованими та мобільними пристроями в персональних мережах (PAN) з використанням дешевих мікросхем-приймачів у кожному пристрої. Технологія працює в діапазоні 2,4 ГГц, ділячи переважаний спектр з іншими технологіями. Швидкість передачі даних становить 3 Мбіт/с, а в версії 3.0 підтримується до 24 Мбіт/с через співрозташоване з'єднання.

Частотний діапазон Bluetooth становить від 2402 МГц до 2480 МГц або від 2400 до 2483,5 МГц з 79 каналами, по 1 МГц на канал. Радіоприймачі Bluetooth перемикаються з одного каналу на інший, використовуючи метод розширення спектра з частотним перестрибуванням (FHSS), що дозволяє уникати зайнятих каналів.

В таблиці 1.3 наведено технічні дані для порівняння бездротових протоколів, які демонструють різні аспекти роботи цих технологій, включаючи діапазон частот, швидкість передачі даних, радіус дії та інші параметри. Ці дані допомагають зрозуміти, як кожен з протоколів може бути використаний у конкретних сценаріях, залежно від потреб користувача та умов експлуатації.

Таблиця 1.3

Порівняння технічних характеристик бездротових протоколів

	<i>ZigBee</i>	<i>WiFi HaLow</i>	<i>Bluetooth</i>	<i>BLE</i>	<i>ANT</i>	<i>Z-Wave</i>	
<i>Standardization</i>	IEEE 802.15.4	IEEE 802.11ah	IEEE 802.15.1	IEEE 802.15.1	Proprietary	Proprietary	
<i>Frequency</i>	2.4 GHz, 868, 915 MHz	900 MHz	2.4 GHz	2.4 GHz	2.4 GHz	900 MHz	
<i>Range, m</i>	10-100	indoor	< 700	1, 10, 100	50	< 30	30
		outdoor	< 1000				
<i>Data rate</i>	20, 40, 250 Kb/s	150-400, 650-780 Kb/s	1, 2, 3 Mb/s	1 Mb/s	1 Mb/s	9.6, 40, 100 Kb/s	
<i>Throughput</i>	10-115.2 Kb/s	> 100 Kb/s	0.7-2.1 Mb/s	305 Kb/s	20 Kb/s	-	
<i>Power consumption, mA</i>	< 40	-	< 30	< 12.5	< 16	< 23	
<i>Tx output power, dBm</i>	from	-3	10	-6	< 19	-20	< 0
	to	10	30	20		0	
<i>Multiplexing</i>	DSSS	OFDM	FHSS	FHSS	TDMA	FHSS	

Протоколи IoT мають значний потенціал для покращення якості життя у розумних будинках, забезпечуючи гнучкість, автоматизацію, енергоефективність та підвищену безпеку. Однак, впровадження цих технологій також пов'язане з певними викликами, такими як ризики для безпеки та конфіденційності, складність

налаштування та управління, сумісність між різними протоколами та високі початкові витрати. Враховуючи ці фактори, важливо ретельно планувати впровадження IoT-технологій, забезпечуючи баланс між перевагами та можливими ризиками для досягнення оптимальних результатів.

2 РОЗРОБКА АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ АВТОМАТИЧНОГО УПРАВЛІННЯ ЕЛЕКТРОПРИЛАДАМИ

2.1 Огляд основних компонентів пристрою

Ми розробляємо систему автоматичного управління електроприладами, яка забезпечить оптимальне освітлення для фікуса в умовах недостатнього природного освітлення. Ця система використовує датчики освітленості та руху для автоматичного контролю включення і виключення світлодіодних ламп. Система, автоматично регулює рівень освітленості, забезпечуючи фікусу необхідні умови для здорового росту, а також знижує енергоспоживання, автоматично вимикаючи освітлення у разі наявності природного світла або відсутності руху в приміщенні. Такий підхід не лише покращує умови для рослин, але й сприяє економії енергії, що є важливим аспектом сучасних технологій. Для виконання всіх своїх задач система має складатися з таких елементів:

1. Центральний контролер (Smart Hub)

Розробка IoT системи автоматичного управління електроприладами потребує вибору надійного та ефективного центрального контролера. У цьому контексті NodeMCU на базі ESP8266(рис 2.1) є оптимальним вибором з кількох технічних та практичних причин.

NodeMCU ESP8266 є платформою для розробки, яка включає Wi-Fi модуль, що забезпечує віддалене керування та моніторинг пристроїв. Це робить його ідеальним для застосування в IoT системах, де бездротове підключення є ключовим аспектом. ESP8266 підтримує стандарт Wi-Fi 802.11 b/g/n, що дозволяє інтеграцію з існуючими мережами без необхідності додаткової інфраструктури.

Іншою важливою характеристикою є багатий набір інтерфейсів та периферійних модулів. ESP8266 підтримує SPI, I2C, UART, PWM, ADC та інші протоколи, що дозволяє легко інтегрувати його з різноманітними датчиками та

виконавчими пристроями. Це робить його дуже гнучким та масштабованим рішенням для IoT систем.

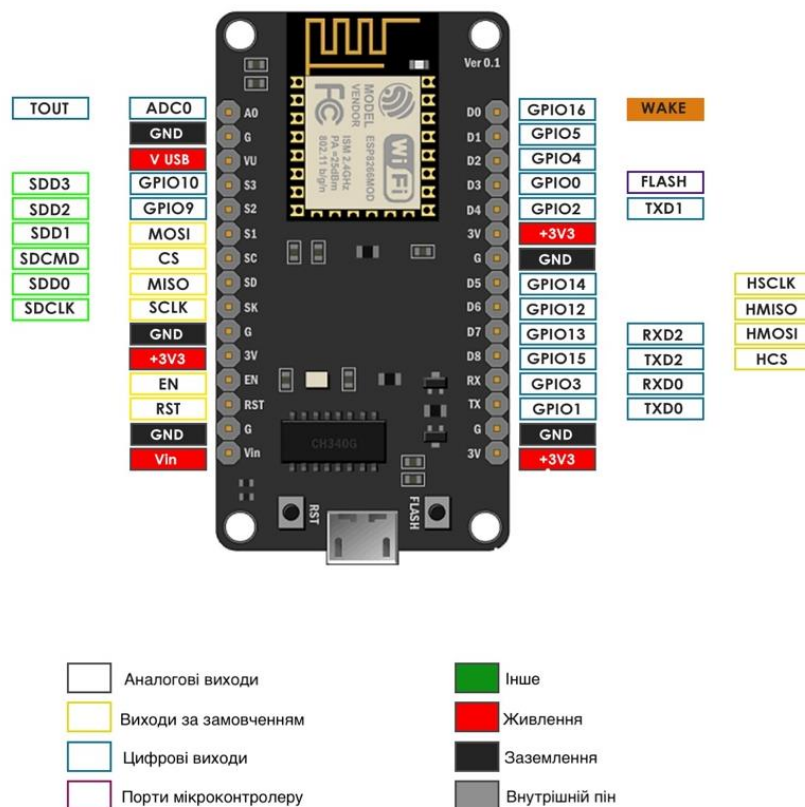


Рис 2.1 Схема NodeMCU ESP8266

Однією з найважливіших переваг ESP8266 є його висока продуктивність та енергоефективність. Процесор Tensilica L106 32-біт RISC працює на частоті 80 або 160 МГц, що забезпечує достатню обчислювальну потужність для обробки даних від численних сенсорів та виконання складних алгоритмів управління. При цьому, ESP8266 має низьке енергоспоживання, що важливо для пристроїв, які працюють від батарей або мають обмежене живлення.

2. Датчик руху

Датчики руху і присутності розташовуються в різних кімнатах будинку і виявляють присутність людей. Для забезпечення ефективної роботи IoT системи автоматичного управління електроприладами найбільш підходящим варіантом є датчик руху HC-SR501 (рис. 2.2). Датчик руху HC-SR501 використовує

інфрачервону технологію для виявлення руху. Він базується на піросенсорі, який реагує на інфрачервоне випромінювання, що випускається тілом людини. HC-SR501 має високу чутливість та широкий діапазон виявлення, що дозволяє йому виявляти рух на відстані до 7 метрів та під кутом до 120 градусів. Ці характеристики роблять його ідеальним для використання у системах автоматичного управління освітленням, безпекових системах та інших IoT застосуваннях.

NodeMCU ESP8266 забезпечує всі необхідні ресурси для обробки даних від датчика руху HC-SR501. Завдяки високій обчислювальній потужності процесора та наявності численних бібліотек для роботи з GPIO пінами, можна легко реалізувати алгоритми обробки сигналів, створення умовних реакцій та інтеграції з іншими компонентами системи.

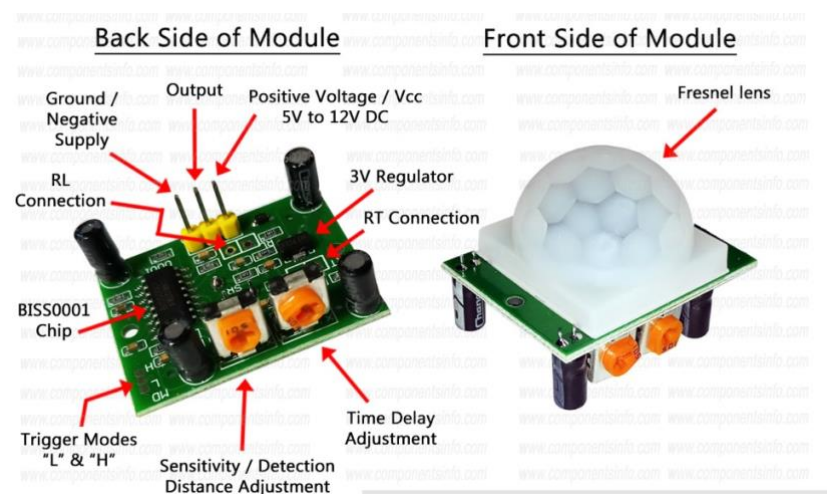


Рис 2.2 Датчик руху HC-SR501 з лицьової та тильної сторони

3. Датчики освітлення

Датчики освітлення вимірюють рівень природного світла в кімнаті.

Одним із найкращих варіантів є використання датчика BH1750. Цей датчик відомий своєю високою точністю та стабільністю вимірювань, що є ключовими параметрами для ефективного управління освітленням в системах розумного будинку.

BH1750(рис 2.3) є цифровим датчиком освітленості, що забезпечує вимірювання інтенсивності світла в люксах (lux). Датчик має високу чутливість і широкий діапазон вимірювань від 1 до 65535 lux, що дозволяє використовувати його як в умовах слабого освітлення, так і при яскравому сонячному світлі.

Основними технічними характеристиками BH1750 є низьке енергоспоживання, що становить близько 0.12 мА в робочому режимі, і компактні розміри, що дозволяють легко інтегрувати його в різноманітні пристрої. Датчик підтримує два адресних режими I2C, що дозволяє використовувати кілька таких датчиків на одній шині I2C. Також він має можливість вибору режимів вимірювання: безперервний або одноразовий, що забезпечує додаткову гнучкість в налаштуванні системи.

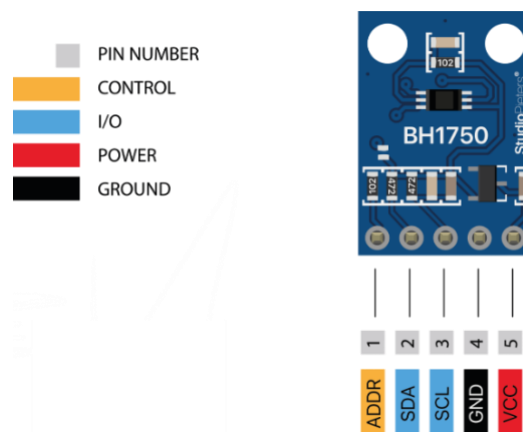


Рис 2.3 Плата датчику освітлення BH1750

Використання BH1750 в системі дозволяє автоматизувати управління освітленням залежно від рівня природного світла, що сприяє енергозбереженню та підвищенню комфорту. Наприклад, система може автоматично вмикати світло в сутінках або знижувати яскравість освітлення в сонячний день, забезпечуючи оптимальні умови освітленості всередині приміщення.

4. Акумулятор

Розробка IoT системи автоматичного управління електроприладами з центральним контролером NodeMCU ESP8266 вимагає вибору надійного джерела

живлення, яке забезпечить стабільну роботу всіх компонентів системи. У цьому контексті літій-іонні акумулятори (Li-ion) є оптимальним вибором завдяки їхнім технічним характеристикам, сумісності та практичним перевагами

Літій-іонний акумулятор мають високу енергоємність, що дозволяє забезпечити тривалу роботу IoT пристроїв без необхідності частого перезарядження. Однією з важливих переваг літій-іонних акумуляторів є їхня низька саморозрядка. Це означає, що акумулятор може зберігати заряд протягом тривалого часу без значних втрат, що є важливим для пристроїв, які працюють в режимі очікування або мають нерегулярний графік використання. Додатково, Li-ion акумулятори мають довгий цикл життя, витримуючи сотні циклів зарядки-розрядки без значного зниження ємності. На рисунку 2.4 можемо розглянути специфікації підібраного акумулятору для нашої системи - NCR18650BF

Cell Type NCR18650BF			
Specifications			
Rated Capacity	*at 20deg.C Min.3200mAh ¹ at 0.2Cdischarge		
Nominal Capacity	*at 25deg.C Min.3250mAh ¹ at 0.2Cdischarge Typ.3350mAh ¹ at 0.2Cdischarge		
Nominal Voltage	3.6V		
Charging Method	Constant Current -Constant Voltage		
Charging Voltage	4.2V		
Charging Current	Std.1625mA		
Charging Time	3.5hrs.		
Ambient Temperature	Charge	0~+45°C	
	Discharge	-20~+60°C	
	Storage	-20~+50°C	
Weight (Max.)	46.5g		
Dimensions (Max.)*	(D)	18.24mm	*without tube
		18.50mm	*with tube
	(H)	65.10mm	*without tube
		65.30mm	*with tube
Volumetric Energy Density	677Wh/l		
Gravimetric Energy Density	248Wh/kg		

TENTATIVE		
Dimensions (Typ.) of Bare Cell	H	65.0mm
	D	18.2mm
	d	9.0 mm

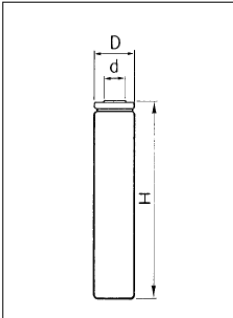


Рис 2.4 Специфікації літій-іонного акумулятору NCR18650BF\

Практичне застосування літій-іонних акумуляторів в IoT системах також включає використання модулів захисту від перевантаження, перенапруги та глибокого розряду, що забезпечує безпеку та довговічність роботи акумулятора. Такі модулі часто інтегровані в готові батарейні блоки, що спрощує їхнє використання для розробників.

5.LEDлампи

Для освітлення рослини було обрано LED стрічку Grow Light USB 5V Однією з головних причин вибору є його висока енергоефективність. Світлодіодні лампи споживають значно менше енергії порівняно з традиційними лампами розжарювання або люмінесцентними лампами, забезпечуючи при цьому достатній рівень освітлення для фотосинтезу. Використання USB-інтерфейсу для живлення дозволяє легко підключити лампу до різноманітних джерел живлення завдяки низькій напрузі живлення (5V) та стандартному інтерфейсу USB, лампу можна безпосередньо підключити до мікроконтролера, що дозволяє автоматично керувати освітленням на основі даних від датчиків. Для загального освітлення я обрав лампу Philips Hue White A19 для загального освітлення вашої системи. Ця лампа підтримує Wi-Fi та Bluetooth для зручного керування через мобільний додаток.

Для інтеграції з NodeMCU ESP8266 можна використовувати шлюз Philips Hue Bridge або інші Zigbee шлюзи, які підтримують MQTT протокол. Завдяки цьому лампу можна легко інтегрувати з іншими розумними пристроями у вашій IoT системі та зручно керувати нею з мобільного додатку або голосовими командами.

2.2 Технічні особливості побудови системи та взаємодії компонентів

Система автоматичного управління освітленням для фікуса побудована на базі мікроконтролера NodeMCU ESP8266, що виконує роль центрального контролера, обробляючи дані від датчиків освітленості та руху та керуючи освітлювальними приладами. Взаємодія компонентів системи забезпечується за допомогою комунікаційних протоколів та електричних з'єднань, що дозволяє створити інтегровану та функціональну систему.

Підключення та взаємодія компонентів

Мікроконтролер NodeMCU ESP8266 підключається до домашньої Wi-Fi мережі, що забезпечує можливість віддаленого моніторингу та керування системою через веб-інтерфейс. Мікроконтролер також забезпечує живлення та комунікацію з датчиками освітленості та руху.

На схемі 2.1 зображено, як датчик освітленості BH1750 підключається до мікроконтролера через інтерфейс I2C, що дозволяє здійснювати точні вимірювання рівня освітленості в реальному часі. Ці дані використовуються для прийняття рішень щодо ввімкнення або вимкнення світлодіодних ламп залежно від умов освітлення.

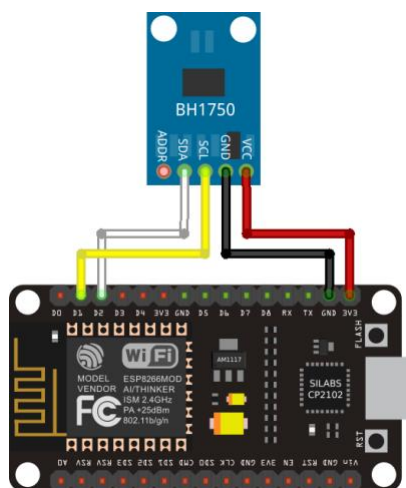


Схема 2.1 Підключення датчику освітлення BH1750 до мікроконтролера NodeMCU ESP8266

Розглядаючи схему 2.2, можемо побачити, підключення датчику руху HC-SR501 до цифрового входу мікроконтролера, що дозволяє виявляти присутність людини в приміщенні. Ці дані використовуються для прийняття рішень щодо ввімкнення або вимкнення світлодіодних ламп залежно від умов освітлення.

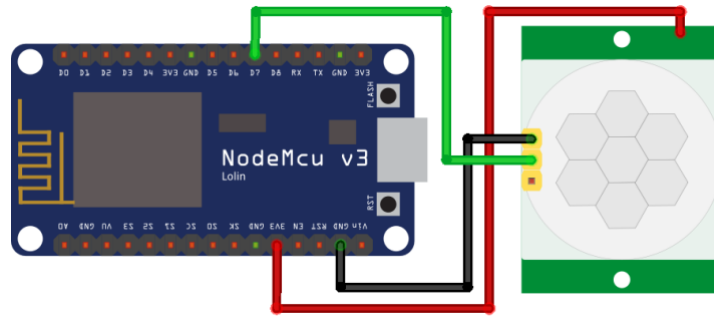


Схема 2.2 Підключення датчику руху HC-SR501 до мікроконтролера NodeMCU ESP8266

Для керування загальною LED лампою використовується реле, яке дозволяє безпечно переключати високовольтне навантаження. Реле підключається до одного з цифрових виходів мікроконтролера та забезпечує розрив або замикання електричного кола живлення великої лампи. Підключення реле та лампи зображено на схемі 2.3, а реалізується таким чином: плюсовий контакт (анод) маленької світлодіодної лампи підключається до цифрового виходу D6 мікроконтролера, мінусовий контакт (катод) підключається до GND мікроконтролера, що забезпечує можливість ввімкнення та вимкнення маленької лампи безпосередньо мікроконтролером. Контакт IN реле підключається до цифрового виходу D7 мікроконтролера, контакти VCC та GND реле підключаються відповідно до 5V та GND джерела живлення, контакт COM реле підключається до фази живлення 220V, а контакт NO (Normally Open) підключається до одного з контактів великої світлодіодної лампи. Другий контакт загальної лампи підключається до нейтрального проводу живлення 220V. Це дозволяє керувати ввімкненням та вимкненням великої лампи через мікроконтролер.

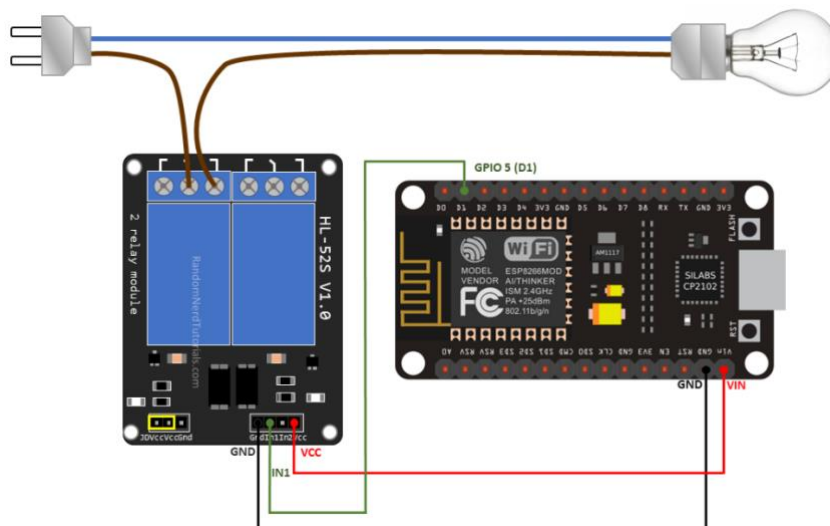


Схема 2.3 Підключення 1-канального реле модулю на 5V та загальної лампи освітлення до мікроконтролера

Для освітлення рослини, такої як фікус, ми можемо використовувати LED Grow Light USB 5V - LED стрічку, яка працює від 5V і може бути підключена безпосередньо до контролера NodeMCU ESP8266. Це рішення є ефективним і зручним, оскільки LED стрічка забезпечує необхідний спектр світла для росту рослин і може живитися від 5V, що відповідає можливостям мікроконтролера. На схемі 2.4 зображено підключення LED стрічки до контролера.

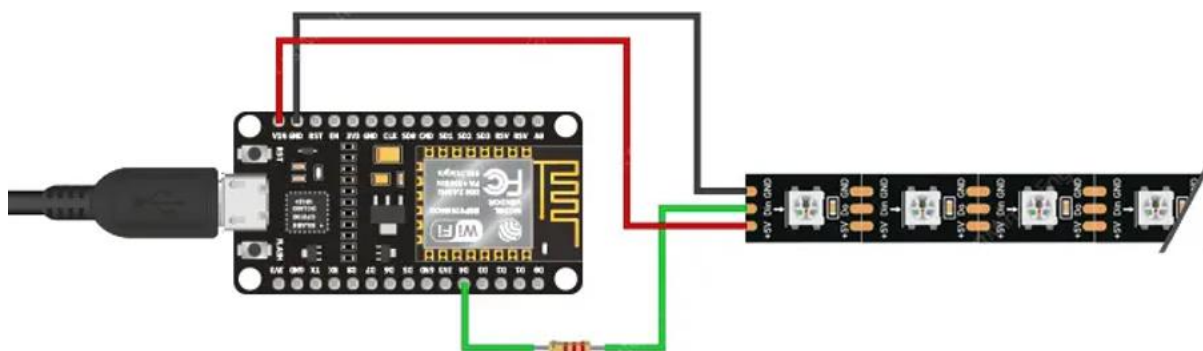


Схема 2.4 Підключення LED стрічки до контролера Node MCU ESP8266

Принципи взаємодії системи

На рисунку 2.5 ми бачимо систему автоматичного управління освітленням

для фікуса в дії. На даний момент датчик руху неактивний, оскільки користувач знаходиться поза зоною дії датчика. Датчик освітленості показує високий рівень освітлення, оскільки за вікном світить сонце, що забезпечує достатнє природне освітлення.

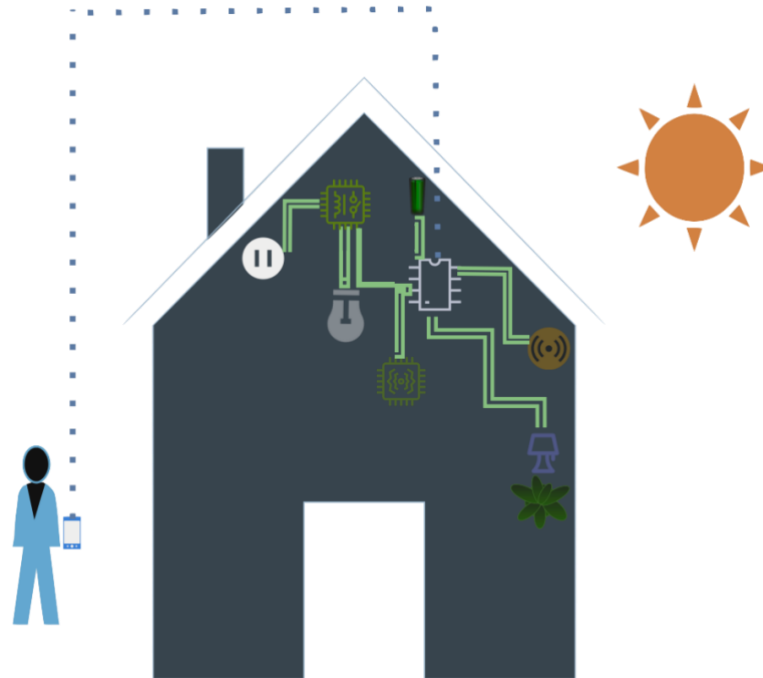


Рис 2.5 Симуляція системи автоматичного контролю електроприладами

Мікроконтролер NodeMCU ESP8266 підключений до домашньої Wi-Fi мережі, що дозволяє користувачеві дистанційно керувати системою через веб-інтерфейс. Датчик освітленості BH1750 підключений до NodeMCU через інтерфейс I2C і вимірює високий рівень освітленості завдяки сонячному світлу, що проникає через вікно. Дані про освітленість передаються на мікроконтролер, який аналізує їх для прийняття рішення про стан освітлення. Датчик руху PIR підключений до цифрового входу NodeMCU і наразі не виявляє руху, оскільки користувач знаходиться поза зоною дії датчика. Сигнал про відсутність руху також передається на мікроконтролер.

5V LED стрічка для рослин підключена до NodeMCU через цифровий вихід і наразі вимкнена, оскільки рівень природного освітлення достатній для фікуса.

Вона живиться від 5V джерела живлення, яке також підключене до NodeMCU. Система автоматично регулює стан освітлення залежно від даних, отриманих від датчиків. Оскільки датчик освітленості BH1750 показує високий рівень освітленості, NodeMCU приймає рішення про вимкнення LED стрічки для рослин, що економить енергію і запобігає зайвому освітленню, коли сонячного світла достатньо. Одночасно датчик руху PIR не виявляє руху в приміщенні, що означає, що користувач не перебуває в зоні дії датчика і немає необхідності вмикати основне освітлення. Таким чином, система залишається в режимі економії енергії, зберігаючи ресурс LED стрічки і використовуючи природне освітлення. На рисунку 2.6 можемо спостерігати позначення для LED стрічки та інших елементів системи.



Рис 2.6 Позначення елементів системи

Через веб-інтерфейс користувач може перевірити поточний стан системи. На веб-сторінці відображаються такі дані: рівень освітленості (високий, що підтверджує достатність сонячного світла), стан датчика руху (неактивний, оскільки руху не виявлено) та стан LED стрічки (вимкнена, оскільки природного освітлення достатньо). Користувач також може вручну керувати LED стрічкою через веб-інтерфейс, наприклад, ввімкнути її для додаткового освітлення, навіть

якщо природного світла достатньо. Це забезпечує додаткову гнучкість у використанні системи.

Отже, на зображенні показана ефективна взаємодія компонентів системи автоматичного управління освітленням для фікуса. Завдяки високому рівню природного освітлення, який виявляє датчик BH1750, та відсутності руху, яку фіксує датчик PIR, система вимикає LED стрічку для економії енергії. Веб-інтерфейс забезпечує зручність дистанційного керування та моніторингу, дозволяючи користувачеві швидко отримати інформацію про стан системи та при необхідності внести корективи в її роботу.

2.3 Налаштування безперебійного електроживлення головного контролера

Забезпечення безперебійного електроживлення є критично важливим для стабільної роботи системи автоматичного управління освітленням, що базується на мікроконтролері NodeMCU ESP8266. Перебої в електропостачанні можуть призвести до втрати даних, збоїв у роботі системи та пошкодження компонентів. У сучасних умовах, коли відключення електроенергії стають дедалі частішими, питання забезпечення безперебійного живлення стає особливо актуальним.

Для створення системи безперебійного живлення необхідні наступні компоненти: акумулятор NCR18650BF, модуль зарядки та захисту акумулятора TP4056[рис.2.7] , блок живлення AMS1117, діод Шотткі та конденсатори для фільтрації пульсацій та стабілізації напруги.

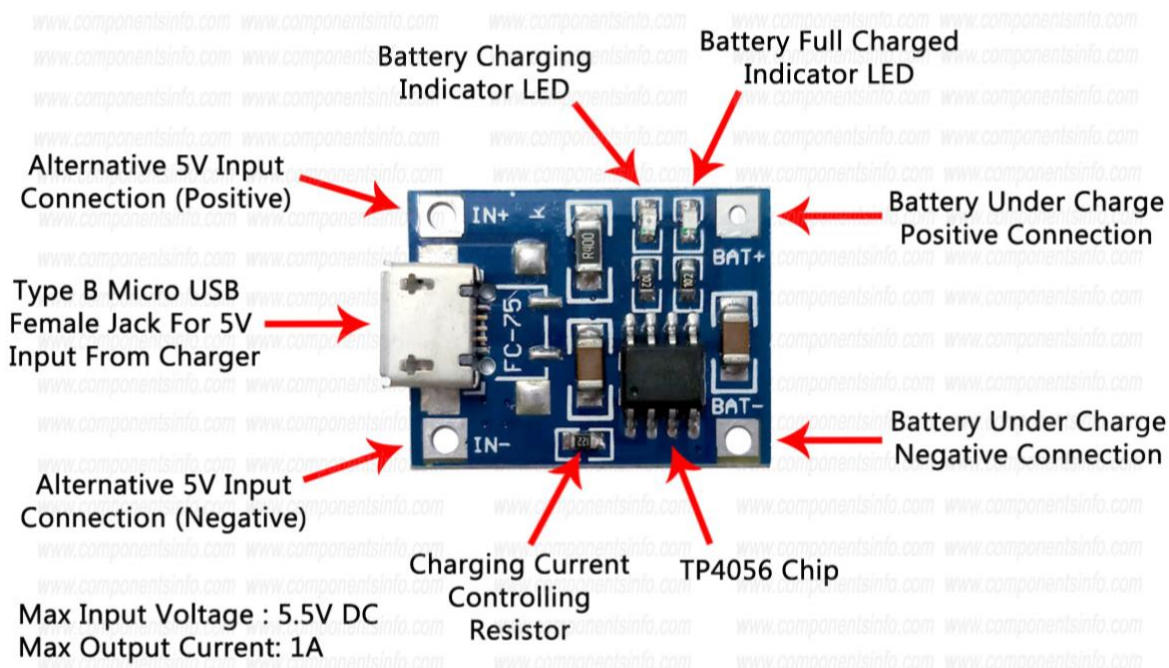


Рис 2.7 Детальний огляд модулю TSP4056

Система безперебійного живлення працює за наступним принципом.

NodeMCU ESP8266 живиться від зовнішнього джерела (USB-адаптер або інший блок живлення на 5V). Модуль TP4056 забезпечує зарядку акумулятора NCR18650BF, контролюючи процес для запобігання перезаряду та перерозряду. У разі відключення основного живлення акумулятор автоматично бере на себе функцію живлення мікроконтролера, забезпечуючи безперебійну роботу системи. Модуль AMS1117 стабілізує напругу, подаючи стабільні 5V на мікроконтролер.

Для забезпечення надійного безперебійного живлення необхідно правильно підключити всі компоненти системи [схема 2.5]. Акумулятор NCR18650BF підключається до модуля зарядки TP4056: BAT+ до позитивного контакту акумулятора, BAT- до негативного контакту акумулятора. Модуль зарядки TP4056 підключається до модуля пониження напруги AMS1117: OUT+ TP4056 до VIN AMS1117, OUT- TP4056 до GND AMS1117. Діод Шоттки підключається між виходом AMS1117 та NodeMCU ESP8266: анод діода до VOUT AMS1117, катод діода до VCC NodeMCU ESP8266. Конденсатори встановлюються на виході AMS1117 між VOUT та GND для фільтрації пульсацій.

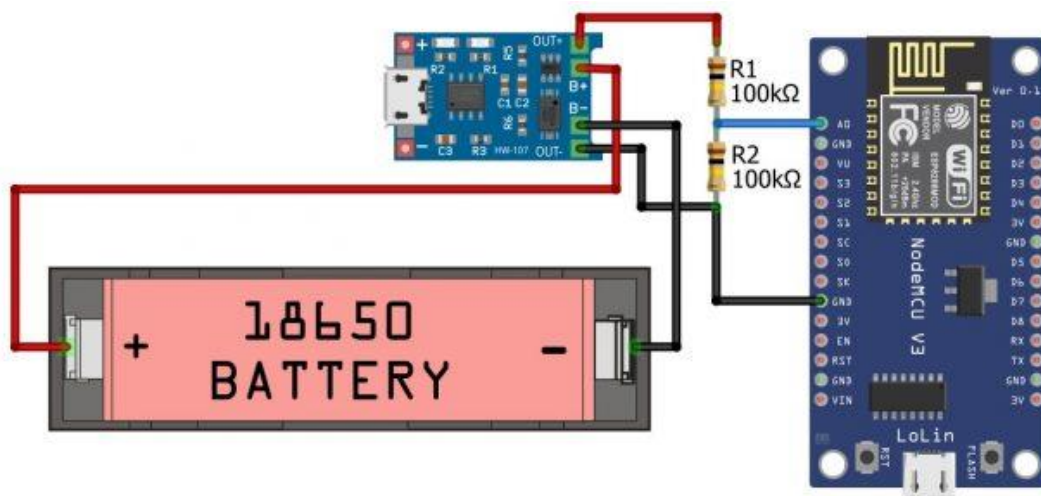


Схема 2.5 Підключення компонентів для безперебійного електроживлення

Запропонована система безперебійного живлення для NodeMCU ESP8266 забезпечує стабільну роботу головного контролера навіть у разі перебоїв у мережі. Використання літій-іонного акумулятора NCR18650BF та модуля зарядки TP4056 дозволяє легко підтримувати акумулятор у зарядженому стані, готовому до використання у будь-який момент. Модуль AMS1117 гарантує стабільне живлення контролера, а діод Шоттки забезпечує безпечне автоматичне перемикання між джерелами живлення. Така конфігурація забезпечує надійну та безперебійну роботу системи автоматичного управління освітленням, що є критично важливим для стабільного функціонування всіх її компонентів.

Важливо зазначити, що сумісність всіх компонентів системи та безперебійна робота датчиків забезпечують ефективне виконання їх функцій. Датчик освітленості BH1750 та датчик руху PIR продовжують коректно працювати навіть під час перемикання між джерелами живлення, завдяки стабільному живленню від акумулятора. У разі відключення основного живлення акумулятор автоматично бере на себе функцію живлення мікроконтролера, забезпечуючи безперебійну роботу системи. Модуль AMS1117 стабілізує напругу, подаючи стабільні 5V на мікроконтролер та всі підключені до нього датчики.

Це дозволяє системі автоматично регулювати освітлення на основі точних даних, отриманих від датчиків, забезпечуючи оптимальні умови для фікуса та підвищуючи загальну ефективність системи. У разі відключення основного живлення акумулятор автоматично бере на себе функцію живлення мікроконтролера, забезпечуючи безперебійну роботу системи. Модуль AMS1117 стабілізує напругу, подаючи стабільні 5V на мікроконтролер та всі підключені до нього датчики.

Розробка апаратної частини для безперебійного живлення є ключовим аспектом у побудові надійної та ефективної системи управління, що гарантує стабільну роботу навіть у разі непередбачених перебоїв у електропостачанні.

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ АВТОМАТИЧНОГО УПРАВЛІННЯ ЕЛЕКТРОПРИЛАДАМИ

3.1 Огляд основних програмних блоків та архітектури програми

Розробка програмного забезпечення для системи автоматичного управління електроприладами є важливим етапом створення ефективної та надійної системи освітлення для фікуса. Цей процес включає в себе інтеграцію різних програмних блоків, що забезпечують стабільну роботу всіх компонентів, ефективну обробку даних від датчиків та надійне керування освітлювальними приладами. Основні програмні блоки системи включають модуль збору даних від датчиків, модуль керування освітленням, веб-сервер та веб-інтерфейс, а також модуль забезпечення безперебійного живлення контролера і датчиків. Кожен з цих блоків відіграє ключову роль у забезпеченні функціональності системи, а їх взаємодія створює цілісну та ефективну архітектуру.

Блок збору даних від датчиків відповідає за зчитування даних з датчика освітленості BH1750 та датчика руху HC-SR501. Нижче наведено приклад коду для реалізації цього блоку на мікроконтролері NodeMCU ESP8266

Архітектура програмного забезпечення системи складається з декількох основних модулів. Першим є модуль збору даних від датчиків, який відповідає за зчитування даних з датчика освітленості BH1750 та датчика руху HC-SR501. Цей модуль проводить аналіз отриманих даних для подальшого прийняття рішень. Другим модулем є модуль керування освітленням, який приймає рішення про ввімкнення або вимкнення світлодіодної стрічки для рослин та великої світлодіодної лампи на основі даних від датчиків. Він забезпечує виконання команд через цифрові виходи мікроконтролера NodeMCU ESP8266. Третій модуль, веб-сервер та веб-інтерфейс, запускається на NodeMCU для забезпечення віддаленого доступу до системи. Цей модуль надає користувачеві інтерфейс для

моніторингу стану системи та ручного керування освітленням, а також відображає поточні дані з датчиків та стан освітлення. Четвертий модуль, модуль забезпечення безперебійного живлення, контролює стан зарядки акумулятора та перемикання між джерелами живлення. Він забезпечує стабільне живлення мікроконтролера та датчиків у разі перебоїв у мережі.

Для кращого розуміння взаємодії між програмними блоками, розглянемо логічну схему, яка відображає всі основні компоненти програмного забезпечення та їх зв'язки. NodeMCU ESP8266 є центральним контролером системи, який запускає веб-сервер для забезпечення віддаленого доступу до системи. Він координує роботу всіх інших модулів, забезпечуючи передачу даних між ними та прийняття відповідних рішень. Модуль збору даних від датчиків включає в себе модуль збору даних з датчика освітленості BH1750, який періодично зчитує рівень освітленості та передає ці дані на NodeMCU, та модуль збору даних з датчика руху HC-SR501, який виявляє наявність руху та передає цифровий сигнал на NodeMCU. Модуль керування освітленням аналізує дані, отримані від модулів збору даних, і на основі цих даних приймає рішення про ввімкнення або вимкнення 5V LED стрічки для рослин та великої світлодіодної лампи через реле.

Веб-сервер та веб-інтерфейс забезпечують віддалений доступ до системи для моніторингу та ручного керування освітленням. Веб-сервер обробляє запити від користувача, дозволяючи йому переглядати стан системи та вручну керувати освітленням через веб-інтерфейс. Модуль забезпечення безперебійного живлення контролює заряд акумулятора NCR18650BF і автоматично перемикає живлення між основним джерелом та акумулятором у разі перебоїв у мережі, забезпечуючи стабільне живлення NodeMCU та підключених датчиків і пристроїв.

Логіка роботи системи полягає в наступному: після запуску NodeMCU підключається до Wi-Fi та запускає веб-сервер. Ініціалізуються всі датчики та модулі. Модуль збору даних з BH1750 періодично зчитує рівень освітленості, а модуль збору даних з HC-SR501 виявляє наявність руху. Модуль керування

освітленням аналізує отримані дані і на основі цих даних приймає рішення про ввімкнення або вимкнення освітлення. Модуль керування освітленням відправляє команди на ввімкнення або вимкнення LED стрічки та великої лампи через реле. Веб-сервер обробляє запити від користувача, дозволяючи йому переглядати стан системи та вручну керувати освітленням через веб-інтерфейс. Модуль забезпечення безперебійного живлення контролює заряд акумулятора та у разі перебоїв у основному живленні автоматично перемикає систему на живлення від акумулятора.

Таким чином, логічна схема відображає взаємодію між програмними блоками системи автоматичного управління електроприладами, показуючи, як дані збираються, обробляються та використовуються для прийняття рішень і керування освітленням, а також як забезпечується віддалений доступ і безперебійне живлення. Це забезпечує стабільну та надійну роботу системи, дозволяючи ефективно управляти освітленням для фікуса та забезпечуючи оптимальні умови для його росту.

3.2 Ключові функції прийому та передачі даних у системі

Розробка програмного забезпечення для системи автоматичного управління електроприладами передбачає створення ефективних функцій для прийому та передачі даних між різними компонентами системи. Це включає отримання даних від датчиків, обробку цих даних, передавання команд для керування освітленням, а також забезпечення взаємодії з користувачем через веб-інтерфейс.

Для забезпечення ефективної роботи системи автоматичного управління електроприладами, яка включає в себе зчитування даних з датчиків, керування освітленням, та взаємодію з користувачем через веб-інтерфейс, необхідно реалізувати ряд ключових функцій. Розглянемо детально код цих функцій і пояснимо кожен рядок, щоб зрозуміти, як вони працюють.

По-перше, розглянемо одну із ключових функцій - зчитування даних з датчика освітленості(рис. 3.1)

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BH1750.h>

Adafruit_BH1750 lightMeter;

void setupLightSensor() {
  Wire.begin(D2, D1); // SDA, SCL
  lightMeter.begin(BH1750::CONTINUOUS_HIGH_RES_MODE);
}

float readLightLevel() {
  return lightMeter.readLightLevel();
}
```

Рис 3.1 Програмний код функції зчитування даних з датчика освітленості

Для зчитування даних з датчика освітленості BH1750 використовується інтерфейс I2C. Функція забезпечує отримання поточних показників освітленості, які використовуються для прийняття рішень про ввімкнення або вимкнення освітлення.

На рисунку 3.2 зображено підключення, бібліотеки Wire, яка дозволяє використовувати протокол I2C для спілкування з датчиками та іншими пристроями, бібліотеки Adafruit Sensor, яка забезпечує уніфікований інтерфейс для роботи з різними датчиками та бібліотеки Adafruit BH1750, яка надає функції для роботи з датчиком освітленості BH1750.

```
#include <Wire.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BH1750.h>
```

Рис 3.2 Підключення бібліотеки Wire, Adafruit Sensor, Adafruit BH1750

Наступним кроком є оголошення об'єкта lightMeter класу Adafruit_BH1750(Рис. 3.3), який представляє датчик освітленості. Це дозволяє нам

використовувати всі методи та функції, надані бібліотекою Adafruit_BH1750, для взаємодії з датчиком.

```
Adafruit_BH1750 lightMeter;
```

Рис 3.3 Оголошення об'єкта lightMeter

Для ініціалізації датчика освітленості використаємо функцію `setupLightSensor()` (Рис. 3.4). Вона викликається один раз при запуску програми. У цій функції ми спочатку ініціалізуємо інтерфейс I2C, використовуючи контакти D2 (SDA) та D1 (SCL) для передачі та прийому даних. Потім ми ініціалізуємо сам датчик BH1750 у режимі безперервного вимірювання з високою роздільною здатністю.

```
void setupLightSensor() {  
  Wire.begin(D2, D1); // SDA, SCL  
  lightMeter.begin(BH1750::CONTINUOUS_HIGH_RES_MODE);  
}
```

Рис 3.4 Ініціалізація датчиків освітленості

Функція `readLightLevel()` (Рис. 3.5) зчитує та повертає поточний рівень освітленості, який вимірює датчик BH1750. Вона викликається для отримання актуальних даних про освітленість та подальшої обробки цих даних у системі.

```
float readLightLevel() {  
  return lightMeter.readLightLevel();  
}
```

Рис 3.5 Програмний код функції `readLightLevel()`

По-друге, розглянемо іншу ключову функцію для зчитування даних з датчика руху PIR, який підключається до цифрового входу мікроконтролера

NodeMCU ESP8266(рис. 3.6). Ця функція зчитує стан датчика (наявність або відсутність руху) і передає ці дані для подальшої обробки.

```
#define PIR_SENSOR_PIN D5

void setupPIRSensor() {
  pinMode(PIR_SENSOR_PIN, INPUT);
}

int readPIRState() {
  return digitalRead(PIR_SENSOR_PIN);
}
```

Рис 3.6 Програмний код ключової функції для зчитування даних з датчика руху

Визначимо пін, до якого підключений датчик руху(рис. 3.7). #define PIR_SENSOR_PIN D5 означає, що ми використовуємо пін D5 на мікроконтролері для підключення датчика руху. Це спрощує подальше використання цього піна в коді.

```
#define PIR_SENSOR_PIN D5
```

Рис 3.7 Назначення піну для датчику руху

Далі, функція setupPIRSensor налаштовує цей пін як вхідний(рис. 3.8). void setupPIRSensor() { pinMode(PIR_SENSOR_PIN, INPUT); } означає, що пін D5 буде використовуватися для зчитування сигналу від датчика руху. pinMode(PIR_SENSOR_PIN, INPUT); встановлює пін D5 як вхідний, що дозволяє мікроконтролеру зчитувати сигнали з датчика.

```
void setupPIRSensor() {
  pinMode(PIR_SENSOR_PIN, INPUT);
}
```

Рис 3.8 Встановлення піна D5, як вхідного

Функція readPIRState зчитує стан датчика руху(рис 3.9). int readPIRState() { return digitalRead(PIR_SENSOR_PIN); } означає, що ця функція зчитує цифровий

сигнал з піна D5 і повертає його значення. `digitalRead(PIR_SENSOR_PIN)` зчитує стан піна (HIGH або LOW), вказуючи на наявність або відсутність руху.

```
int readPIRState() {
    return digitalRead(PIR_SENSOR_PIN);
}
```

Рис 3.9 Програмний код функції `readPIRState()`

Також, розглянемо третю ключову функцію - функцію керування освітленням, яка приймає рішення про ввімкнення або вимкнення світлодіодної стрічки на основі даних, отриманих від датчиків(рис. 3.10)

```
#define LED_PIN D6

void setupLED() {
    pinMode(LED_PIN, OUTPUT);
    digitalWrite(LED_PIN, LOW);
}

void controllighting(float lux, int pirState) {
    if (lux < 50 && pirState == LOW) {
        digitalWrite(LED_PIN, HIGH); // Включити LED стрічку
    } else {
        digitalWrite(LED_PIN, LOW); // Вимкнути LED стрічку
    }
}
```

Рис 3.10 Програмний код ключової функції керування освітленням

Перша частина коду визначає пін, до якого підключена світлодіодна стрічка(рис. 3.11). `#define LED_PIN D6` означає, що ми використовуємо пін D6 на мікроконтролері для підключення світлодіодної стрічки.

```
#define LED_PIN D6
```

Рис 3.11 Визначення пін до якого підключення світлодіодна стрічка

Функція `setupLED` налаштовує цей пін як вихідний (рис. 3.12). `void setupLED() { pinMode(LED_PIN, OUTPUT); digitalWrite(LED_PIN, LOW); }` означає, що пін D6 буде використовуватися для керування світлодіодною стрічкою. `pinMode(LED_PIN, OUTPUT);` встановлює пін D6 як вихідний, а `digitalWrite(LED_PIN, LOW);` вимикає світлодіодну стрічку при початковій ініціалізації.

```
void setupLED() {
  pinMode(LED_PIN, OUTPUT);
  digitalWrite(LED_PIN, LOW);
}
```

Рис 3.12 Програмний код підфункції `setupLED`

Функція `controlLighting` (рис 3.13) приймає два параметри: `lux` (рівень освітленості) та `pirState` (стан датчика руху). Вона приймає рішення про ввімкнення або вимкнення світлодіодної стрічки. `if (lux < 50 && pirState == LOW) { digitalWrite(LED_PIN, HIGH); } else { digitalWrite(LED_PIN, LOW); }` означає, що якщо рівень освітленості нижче 50 і немає руху, світлодіодна стрічка вмикається, інакше вимикається.

```
void controlLighting(float lux, int pirState) {
  if (lux < 50 && pirState == LOW) {
    digitalWrite(LED_PIN, HIGH); // Включити LED стрічку
  } else {
    digitalWrite(LED_PIN, LOW); // Вимкнути LED стрічку
  }
}
```

Рис 3.13 Програмний код підфункції `controlLighting`

Останньої ключовою функцією є функція веб-сервера для взаємодії з користувачем (рис. 3.14). Веб-сервер на NodeMCU ESP8266 забезпечує взаємодію з користувачем через веб-інтерфейс.

```

#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>

ESP8266WebServer server(80);

void setupWiFi() {
  WiFi.begin("your_SSID", "your_PASSWORD");
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
  }
}

void setupWebServer() {
  server.on("/", handleRoot);
  server.on("/status", handleStatus);
  server.on("/toggleLed", handleToggleLed);
  server.begin();
}

void handleRoot() {
  String html = "<html><head><title>Управління освітленням</title></head><body>";
  html += "<h1>Управління освітленням</h1>";
  html += "<p><a href=\"/status\">Статус системи</a></p>";
  html += "<p><a href=\"/toggleLed\">Переключити лампу</a></p>";
  html += "</body></html>";
  server.send(200, "text/html", html);
}

void handleStatus() {
  String status = "{\"lux\":";
  status += readLightLevel();
  status += ", \"pir\":";
  status += readPIRState();
  status += ", \"led\":";
  status += digitalRead(LED_PIN);
  status += "}";
  server.send(200, "application/json", status);
}

void handleToggleLed() {
  int state = digitalRead(LED_PIN);
  digitalWrite(LED_PIN, !state);
  server.send(200, "text/html", "<p>Лампа переключена</p><p><a href=\"/\">Назад</a>");
}

void loop() {
  server.handleClient();
}

```

Рис 3.14 Програмний код ключової функції веб-сервера для взаємодії з користувачем

Спочатку ми підключаємо необхідні бібліотеки для роботи з Wi-Fi та веб-сервером(рис 3.15): `#include <ESP8266WiFi.h>` та `#include <ESP8266WebServer.h>`. Ці бібліотеки забезпечують необхідний інструментарій для налаштування бездротового з'єднання та створення простого веб-сервера.

```

#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>

```

Рис 3.15 Підключення бібліотеки для роботи з Wi-Fi та веб-сервером

Далі, створюється об'єкт `server` класу `ESP8266WebServer`(рис 3.16), який представляє веб-сервер, що отримає сигнал на порту 80.

```

ESP8266WebServer server(80);

```

Рис 3.16 Створення об'єкту `server`

На рисунку 3.17, зображена функція `setupWiFi` підключає мікроконтролер до Wi-Fi мережі, в таблиці 3.1 знаходиться лістинг функції `setupWiFi`

```
void setupWiFi() {
  WiFi.begin("your_SSID", "your_PASSWORD");
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
  }
}
```

Рис 3.17 Програмний код підфункції `setupWiFi()`

Таблиця 3.1

Лістинг функції `setupWiFi`

Рядок коду	Призначення
<code>void SetupWiFi() {</code>	Початок функції, її запуск
<code>WiFi.begin("your_SSID", "your_PASSWORD");</code>	Початок підключення до Wi-Fi мережі
<code>while (WiFi.status() != WL_CONNECTED)</code>	Очікування підключення до Wi-Fi
<code>delay(1000) }</code>	Затримка в 1 секунду між спробами підключення
<code>}</code>	Завершення функції

Функція `setupWebServer` (рис. 3.18) налаштовує веб-сервер. В неї входить обробка запиту для головної сторінки, визначає обробника запиту для сторінки статусу системи та обробника запиту для перемикання стану лампи, а також запуск веб-серверу. В таблиці 3.2 написаний лістинг коду функції.

```
void setupWebServer() {
  server.on("/", handleRoot);
  server.on("/status", handleStatus);
  server.on("/toggleLed", handleToggleLed);
  server.begin();
}
```

Рис 3.18 Функція налаштування веб-сервера

Таблиця 3.2

Лістинг функції setupWebServer()

Рядок коду	Призначення
void setupWebServer() {	Початок функції, її запуск
server.on("/", handleRoot);	Визначення обробника для головної сторінки
server.on("/status", handleStatus);	Визначення обробника для сторінки статусу системи
server.on("/toggleLed", handleToggleLed);	Визначення обробника для перемикачя стану лампи
server.begin();	Запуск веб-сервера
}	Закінчення функції

Функція handleRoot(рис 3.19) відправляє користувачеві HTML-сторінку з головного інтерфейсу управління освітленням, створює HTML-код сторінки та відправляє цю сторінку користувачеві, що відображається у веб-браузері користувача, і забезпечує можливість взаємодії з системою через веб-інтерфейс.

```
void handleRoot() {
  String html = "<html><head><title>Управління освітленням</title></head><body>";
  html += "<h1>Управління освітленням</h1>";
  html += "<p><a href=\"/status\">Статус системи</a></p>";
  html += "<p><a href=\"/toggleLed\">Переключити лампу</a></p>";
  html += "</body></html>";
  server.send(200, "text/html", html);
}
```

Рис 3.19 Програмний код функції handleRoot()

В таблиці 3.3 написаний лістинг програмного коду підфункції handleRoot()

Таблиця 3.3

Лістинг програмного коду функції handleRoot()

Рядок коду	Призначення
void handleRoot() {	Початок функції, її запуск
html += "<h1>Управління освітленням</h1>"; html += "<p>Статус системи</p>"; html += "<p>Переключити лампу</p>"; html += "</body></html>";	Створює HTML-код сторінки Абзаци, посилання, заголовки
server.send(200, "text/html", html);	Відправка HTML-коду користувачеві
}	Завершення функції

Функція `handleStatus`(рис 3.20) відправляє JSON-дані про поточний стан системи, створює JSON-рядок з рівнем освітленості, станом датчика руху та станом світлодіодної стрічки та відправляє ці дані користувачеві. Розглянемо детально кожен аспект цієї функції та створення JSON-рядка в таблиці 3.4.

JSON (JavaScript Object Notation) — це легкий формат обміну даними, який легко читається і записується як людиною, так і машиною. JSON-об'єкти складаються з пар "ключ-значення", що дозволяє легко організувати та структурувати інформацію. У нашій системі JSON використовується для передачі даних про рівень освітленості, стан датчика руху та стан світлодіодної стрічки.

```
void handleStatus() {
    String status = "{\"lux\":";
    status += readLightLevel();
    status += ", \"pir\":";
    status += readPIRState();
    status += ", \"led\":";
    status += digitalRead(LED_PIN);
    status += "}";
    server.send(200, "application/json", status);
}
```

Рис 3.20 Програмний код підфункції `handleStatus()`

Таблиця 3.4

Лістинг програмного коду функції `handleStatus`

Рядок коду	Призначення
<code>void handleStatus() {</code>	Початок функції, її запуск
<code>String status = "{\"lux\":";</code>	Оголошення змінної <code>status</code> типу <code>String</code> та ініціалізація її значенням.
<code>status += readLightLevel();</code>	Виклик функції <code>readLightLevel()</code> , яка зчитує поточний рівень освітленості.
<code>status += ", \"pir\":";</code>	Додавання до JSON-рядка <code>status</code> нового поля "pir", яке буде містити стан датчика руху.
<code>status += readPIRState();</code>	Виклик функції <code>readPIRState()</code> , яка зчитує поточний стан датчика руху
<code>status += ", \"led\":";</code>	Додавання до JSON-рядка <code>status</code> нового поля "led", яке буде містити стан світлодіодної стрічки.

Продовження таблиці 3.4

<code>status += digitalRead(LED_PIN);</code>	Виклик функції <code>digitalRead(LED_PIN)</code> , яка зчитує стан світлодіодної стрічки (ввімкнена або вимкнена).
<code>status += "},";</code>	Додавання закриваючої дужки <code>}</code> до JSON-рядка <code>status</code> , завершуючи формування JSON-структури.
<code>server.send(200, "application/json", status);</code>	Відправка HTTP-відповіді клієнту. Перший аргумент <code>200</code> вказує на статус-код HTTP-відповіді (<code>200</code> означає "ОК"). Другий аргумент встановлює MIME-тип відповіді, що вказує на формат даних JSON. Третій аргумент <code>status</code> містить дані про стан системи.
<code>}</code>	Завершення функції

Функція `handleToggleLed` (рис. 3.21) виконує важливу роль в системі автоматичного управління освітленням, забезпечуючи інтерактивне керування світлодіодною стрічкою через веб-інтерфейс. Її основна мета полягає в тому, щоб дозволити користувачам дистанційно змінювати стан освітлення з використанням простого веб-браузера. Коли користувач надсилає запит на зміну стану світлодіодної стрічки, функція зчитує поточний стан стрічки, інвертує його, тобто перемикає з увімкненого стану на вимкнений або навпаки, і потім відправляє користувачеві повідомлення, яке підтверджує зміну стану освітлення.

```
void handleToggleLed() {
    int state = digitalRead(LED_PIN);
    digitalWrite(LED_PIN, !state);
    server.send(200, "text/html", "<p>
}
```

Рис 3.21 Програмний код функції `handleToggleLed`

Функція також забезпечує надійний зворотний зв'язок для користувача, відправляючи підтвердження про успішне виконання команди через веб-інтерфейс. Це підтвердження включає HTML-код, який відображається в браузері користувача, що дає змогу легко побачити результат своєї дії. Таким чином, функція **handleToggleLed** виконує ключову роль в інтерактивній частині системи управління освітленням, роблячи її більш зручною та ефективною для кінцевих користувачів. Її реалізація демонструє, як сучасні веб-технології можуть бути інтегровані з фізичними пристроями для створення інтуїтивно зрозумілих та легко керованих систем. Детальне пояснення кожного рядка коду цієї функції наведено в таблиці 3.5

Таблиця 3.5

Лістинг програмного коду функції `handleToggleLed()`

Рядок коду	Призначення
<code>void handleToggleLed() {</code>	Початок функції, її запуск
<code>int state = digitalRead(LED_PIN);</code>	Зчитування поточного стану світлодіодної стрічки. Функція <code>digitalRead(LED_PIN)</code> зчитує цифровий сигнал з пінa <code>LED_PIN (D6)</code> . Значення буде <code>HIGH (1)</code> , якщо світлодіодна стрічка увімкнена, або <code>LOW (0)</code> , якщо вимкнена. Результат зчитування зберігається в змінну <code>state</code> .
<code>digitalWrite(LED_PIN, !state);</code>	Перемикання стану світлодіодної стрічки.
<code>server.send(200, "text/html", "<p>Лампа переключена</p><p>Назад</p>");</code>	Відправка відповіді користувачеві через веб-сервер. Функція <code>server.send</code> відправляє HTTP-відповідь з кодом статусу 200 (OK) та вмістом типу <code>text/html</code> . Відправляється HTML-код, який містить повідомлення про успішне перемикання стану лампи і посилання для повернення на головну сторінку веб-інтерфейсу
<code>}</code>	Закриття визначення функції.

Функції зчитування даних від датчиків, керування освітленням та веб-сервер працюють разом для забезпечення стабільної та ефективної роботи системи(рис. 3.22). Усі функції інтегруються у основний цикл програми, забезпечуючи безперервний моніторинг стану датчиків, автоматичне керування освітленням та взаємодію з користувачем.

```
void setup() {
  setupLightSensor();
  setupPIRSensor();
  setupLED();
  setupWiFi();
  setupWebServer();
}

void loop() {
  float lux = readLightLevel();
  int pirState = readPIRState();
  controllLighting(lux, pirState);
  server.handleClient();
  delay(500); // Затримка для зменшення частоти читання
}
```

Рис 3.22 Інтеграція всіх ключових функцій програми

Функція `setup` викликається один раз при запуску програми і налаштовує всі необхідні компоненти: датчик освітленості, датчик руху, світлодіодну стрічку, Wi-Fi та веб-сервер. Функція `loop` викликається безперервно і забезпечує зчитування даних від датчиків, керування освітленням та обробку запитів від веб-сервера.

Таким чином, ключові функції прийому та передачі даних у системі автоматичного управління електроприладами забезпечують зчитування даних від датчиків, обробку цих даних, керування освітленням та взаємодію з користувачем через веб-інтерфейс. Інтеграція цих функцій у єдину систему дозволяє створити стабільну та ефективну систему, яка автоматично регулює освітлення на основі отриманих даних, забезпечуючи оптимальні умови для рослини та зручність для користувача.

3.3 Базові принципи досягнення надійності програмного забезпечення у вашій програмі

Надійність програмного забезпечення є ключовим аспектом для забезпечення безперебійної роботи системи автоматичного управління електроприладами. У моїй програмі, яка реалізує систему управління освітленням на базі NodeMCU ESP8266, застосовуються кілька ключових принципів для досягнення високої надійності.

Одним з основних принципів є модульність, яка передбачає розбиття програми на окремі незалежні модулі. У моїй програмі це реалізовано через функції, такі як `setup()`, `loop()`, `handleStatus()`, `handleToggleLed()`, що забезпечують різні аспекти роботи системи. Кожен модуль виконує свою конкретну функцію, що полегшує тестування, обслуговування та оновлення програми. Іншим важливим принципом є інкапсуляція, яка передбачає приховування внутрішніх деталей реалізації функцій. Наприклад, функції `readLightLevel()` і `readPIRState()` інкапсулюють логіку зчитування даних від датчиків, що дозволяє використовувати їх без необхідності знати деталі їх реалізації. Це зменшує залежність між компонентами та робить систему більш стійкою до змін.

Принцип єдиної відповідальності також відіграє важливу роль у моїй програмі. Він реалізований через чіткий поділ функцій, де кожна функція відповідає за виконання лише однієї задачі. Наприклад, функція `handleStatus()` відповідає лише за формування та відправку статусу системи, а функція `handleToggleLed()` — за перемикання стану світлодіодної стрічки. Такий підхід спрощує розробку, тестування та обслуговування коду.

Важливим аспектом забезпечення надійності є обробка виключень. У моїй програмі передбачено обробку можливих помилок, таких як збої при зчитуванні даних від датчиків або підключення до Wi-Fi. Наприклад, додано перевірки на успішне підключення до Wi-Fi у функції `setup()`, яка повторює спробу підключення

кожні 1000 мс, доки з'єднання не буде встановлено. Це дозволяє мінімізувати ризик відмови системи та забезпечити її безперебійну роботу. Тестування є критично важливим для забезпечення надійності програмного забезпечення. У моїй програмі проводиться модульне тестування основних функцій, таких як `readLightLevel()`, `readPIRState()`, щоб гарантувати їх правильну роботу в різних умовах.

Безпека програмного забезпечення є ще одним важливим принципом, реалізованим у моїй програмі. Забезпечення безпеки включає виявлення та усунення вразливостей, захист даних та контроль доступу. Наприклад, доступ до веб-інтерфейсу обмежується паролем, що запобігає несанкціонованому доступу до системи управління освітленням. Реалізація цього здійснена через додавання базової аутентифікації у функцію `setup()`, яка перевіряє облікові дані користувача перед наданням доступу до інтерфейсу управління.

Моніторинг та логування є важливими для підтримки надійності та швидкого реагування на проблеми. У моїй програмі реалізовано функції логування основних подій та стану системи, що дозволяє відстежувати роботу системи в реальному часі та оперативно виявляти потенційні проблеми. Наприклад, кожен запит до веб-сервера та стан датчиків фіксуються у лог-файлах для подальшого аналізу. Логування реалізовано через додавання функції `logEvent()`, яка записує важливі події та стани у файли на сервері.

Резервне копіювання та відновлення даних також забезпечують надійність системи. У моїй програмі передбачено регулярне створення резервних копій конфігураційних файлів та налаштувань системи. Це дозволяє швидко відновити роботу системи у разі виникнення збоїв або втрати даних. Наприклад, резервне копіювання налаштувань системи виконується автоматично кожні 24 години за допомогою функції `backupSettings()`, яка зберігає копії конфігураційних файлів у безпечному місці.

Застосування принципів модульності, інкапсуляції, єдиної відповідальності, обробки виключень, тестування, безпеки, моніторингу та логування, а також резервного копіювання і відновлення дозволяє забезпечити високу надійність програмного забезпечення. Це, у свою чергу, забезпечує безперебійну роботу системи автоматичного управління електроприладами, що є критичним для досягнення стабільної та ефективної роботи всієї системи. Надійне програмне забезпечення мінімізує ризик відмов, забезпечує стабільну роботу та задоволення потреб користувачів, а також сприяє ефективному обслуговуванню та подальшому розвитку системи.

ВИСНОВОК

Говорячи про автоматизацію управління побутовими електроприладами за допомогою IoT, необхідно розуміти, що це – сучасний підхід до оптимізації використання електроенергії та підвищення комфорту життя. Інтернет речей (IoT) дозволяє об'єднувати різноманітні пристрої в єдину мережу, забезпечуючи їх взаємодію та централізоване управління через мобільні додатки або голосові асистенти, такі як Alexa, Google Home та SmartThings.

У світлі останніх тенденцій до створення розумних будинків, автоматизація побутових процесів є важливим кроком на шляху до підвищення енергоефективності та зручності використання електроприладів. Використання IoT-систем для автоматичного управління освітленням, безпековими системами та іншими аспектами побуту дозволяє значно знизити споживання електроенергії та забезпечити комфортне проживання. Дослідження сучасних технологій IoT показало, що з кожним роком кількість розумних пристроїв збільшується, а їх вартість знижується, що робить їх доступнішими для широкого кола користувачів.

Дослідження сучасних платформ для автоматизації, таких як Alexa, Google Home та SmartThings, показало їх високу ефективність та зручність використання. Ці платформи дозволяють легко інтегрувати різноманітні пристрої у єдину систему, забезпечуючи централізоване управління та моніторинг. Важливою перевагою таких систем є можливість налаштування автоматичних сценаріїв, які підвищують ефективність використання електроприладів та забезпечують комфортні умови проживання.

Під час вивчення принципу роботи та використання розробленої системи автоматичного управління побутовими електроприладами були виявлені такі переваги: низька вартість, легкість програмування, висока інтеграція з іншими пристроями та платформами, гнучкість у налаштуванні та розширенні функціональності, можливість автономної роботи та віддаленого управління. До недоліків використання таких систем можна віднести необхідність регулярного

технічного обслуговування та налаштування, потенційні ризики безпеки, пов'язані з використанням мережевих технологій, а також неможливість повноцінної роботи від акумулятора, доступний тільки обмежений функціонал, оскільки загальна лампа споживає занадто багато енергії.

Проте переваги таких систем значно перевищують їх недоліки. Використання IoT для автоматизації управління побутовими електроприладами дозволяє не лише знизити витрати на електроенергію, але й забезпечити високу якість життя, комфорт та безпеку у домі. Розроблена система є важливим кроком у напрямку створення розумного будинку, який здатний самостійно адаптуватися до потреб користувача та забезпечувати оптимальні умови проживання.

Практична значущість результатів дослідження полягає у можливості використання розробленої системи для покращення якості освітнього процесу в університетах, надання студентам можливості здобути нові практичні навички з розробки та роботи з подібними системами. Система також може використовуватися у реальних умовах для автоматизації догляду за кімнатними рослинами, забезпечуючи оптимальні умови для їх росту та розвитку.

Апаратна частина системи складається з таких основних компонентів: датчик освітленості та руху (BH1750 та HC-SR501), світлодіодна стрічка для підсвічування рослин, розумна лампа, реле для керування великою лампою, акумулятор та плата керування NodeMCU ESP8266.

Датчик освітленості та руху слугує для вимірювання рівня освітленості та виявлення руху, що дозволяє визначати необхідність ввімкнення або вимкнення освітлення. Світлодіодна стрічка забезпечує підсвічування рослин, а розумна лампа використовується для основного освітлення приміщення. Реле дозволяє безпечно вмикати та вимикати високопотужні електроприлади за допомогою низьковольтних сигналів від мікроконтролера, забезпечуючи надійне та ефективне управління загальною лампою.

Програмне забезпечення системи діє за таким алгоритмом: зчитуються показники датчика освітленості та руху, після чого проводяться розрахунки та

приймаються рішення щодо ввімкнення або вимкнення освітлювальних приладів. Усі дані передаються до веб-інтерфейсу або мобільного додатка для віддаленого моніторингу та керування системою.

Однією з причин, чому була обрана плата NodeMCU ESP8266 для управління системою, стало те, що вона має вбудований Wi-Fi чіпсет. Це дозволяє використовувати дистанційне управління пристроєм по мережі. Проект має потенціал, існує можливість додати велику кількість датчиків та розумних пристроїв для автоматизації.


ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Sang Guun Yoo. Architecture of the Implemented System Amazon Services. ResearchGate. URL: https://www.researchgate.net/figure/Architecture-of-the-Implemented-System_fig1_330488655.
2. Alexa skills - Serverless Applications Lens. URL: <https://docs.aws.amazon.com/wellarchitected/latest/serverless-applications-lens/alexaskills.html>.
3. Harmony and Amazon Alexa. Logitech Harmony Support. URL: <https://support.myharmony.com/en-us/harmony-experience-with-amazon-alexa>.
4. Luca Calderoni. Samsung SmartThings architecture. ResearchGate, URL: https://www.researchgate.net/figure/Samsung-SmartThings-architecture-23_fig4_332532551.
5. Samsung Home API Documentation. DeveloperSmartThings. URL: <https://developer.smarthings.com/docs/home-api/get-started-with-home-api>.
6. Developers Home Google. URL: <https://developers.home.google.com/cloud-to-cloud/primer/home-graph>.
7. Alex Akinbi. Google Assistant ecosystem. ResearchGate. URL: https://www.researchgate.net/figure/Google-Assistant-ecosystem-a-paradigm-for-smart-home-IoT-forensic-ecosystem_fig4_343658076.
8. Sustainability in New and Emerging Technologies - Welcome to 6GWorld. *6GWorld*. URL: <https://www.6gworld.com/projects/sustainability-in-new-and-emerging-technologies/>.
9. Deepti Sharma, Ginni Nijhawan. The Economic Viability of Smart Home Investments: A Cost- Benefit Analysis. *BIO Web of Conferences*. URL: https://www.bio-conferences.org/articles/bioconf/pdf/2024/05/bioconf_rtbs2024_01086.pdf.

10. Oleh Horyachyy. Comparison of Wireless Communication Technologies used in a Smart Home. *Diva Portal*. URL: <http://www.diva-portal.org/smash/get/diva2:1118965/FULLTEXT02.pdf>.
11. ESP8266 [Електронний ресурс] – Режим доступу до ресурсу: <https://en.wikipedia.org/wiki/ESP8266>.
12. ESP8266 Pinout Reference: Which GPIO pins should you use? [Електронний ресурс] – Режим доступу до ресурсу: <https://randomnerdtutorials.com/esp8266-pinout-reference-gpios/>.
13. К.П. Сторчак, А.М. Тушич, І.М. Срібна, Н.Д. Яковенко, Д.В. Кравець «Технології Інтернет речей». - 2021.
14. Положення про кваліфікаційну роботу бакалавра: методичні вказівки до виконання та захисту кваліфікаційної роботи бакалавра для студентів денної та заочної форм. –К.: ДУТ, 2024. – 32 с.
15. Babita Rawat. Energy Efficiency Assessment in Smart Homes: A Comparative Study of Energy Efficiency Tests. *ResearchGate*. URL: https://www.researchgate.net/publication/377359808_Energy_Efficiency_Assessment_in_Smart_Homes_A_Comparative_Study_of_Energy_Efficiency_Tests.

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ

ПРЕЗЕНТАЦІЯ ДО ЗАХИСТУ БАКАЛАВРСЬКОЇ РОБОТИ
 СТУДЕНТА ГРУПИ ІСД-42
 СПЕЦІАЛЬНОСТІ 126 ІНФОРМАЦІЙНІ СИСТЕМИ ТА
 ТЕХНОЛОГІЇ
 СТРЕЛЬЧЕНКА ВЛАДИСЛАВА



**Розробка IoT- системи для
 автоматичного управління
 побутовими електроприладами**

1

МЕТА, ОБ`ЄКТ І ПРЕДМЕТ РОБОТИ

Об`єкт дослідження - процес автоматизації управління побутовими електроприладами з використанням технологій Інтернету речей (IoT) для підвищення енергоефективності, підвищення зручності та комфорту у домашніх умовах.

Предмет дослідження – способи розробки та використання IoT-систем для автоматичного управління побутовими електроприладами, які можна використовувати для догляду за кімнатними рослинами та в якості навчального макета для лабораторії університету.

Мета роботи – дослідження сучасних технологій Інтернету речей (IoT) та автоматизації побутових процесів, розробка концепції IoT-системи для автоматичного управління побутовими електроприладами з використанням датчика освітленості та руху, розумної лампи та мікроконтролера NodeMCU ESP8266, яка може використовуватися для догляду за кімнатними рослинами

2

Поняття ІОТ в контексті розумного будинку

Інтернет речей (IoT, Internet of Things) в контексті розумного будинку означає систему взаємопов'язаних електронних пристроїв, сенсорів та програмного забезпечення, які здатні збирати та обмінюватися даними через інтернет або інші мережі. Ці пристрої можуть взаємодіяти один з одним та з користувачами, автоматизуючи різноманітні процеси в будинку для підвищення комфорту, безпеки та енергоефективності.



SmartThings використовує хаб як центральний контролер, який об'єднує різні розумні пристрої в єдину систему. Користувачі можуть керувати своїми пристроями через мобільний додаток SmartThings, налаштовувати автоматичні сценарії та отримувати сповіщення. Основні переваги: висока сумісність з широким спектром пристроїв, централізоване управління, гнучкість налаштувань та зручність використання. SmartThings забезпечує інтеграцію та автоматизацію всіх аспектів розумного будинку.

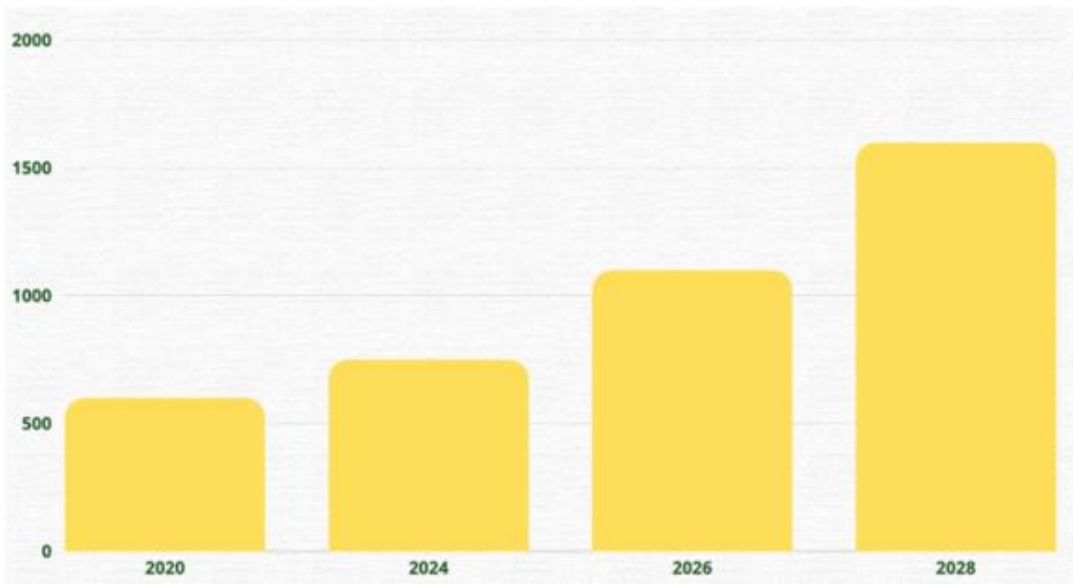
3



Сучасні системи управління електроприладами

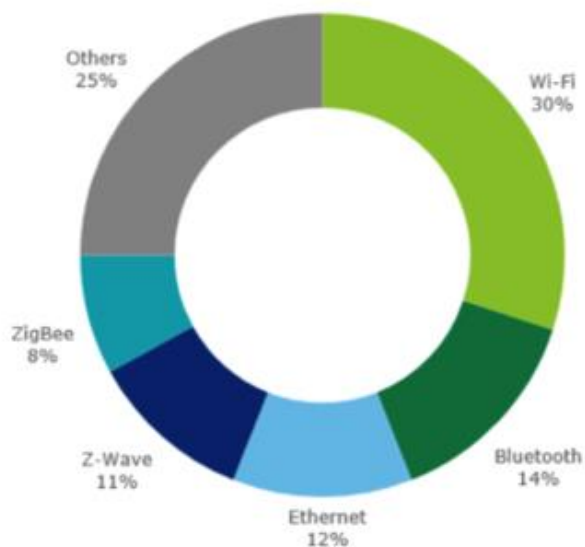
Amazon Alexa є однією з найпопулярніших систем управління розумним будинком. Вона дозволяє користувачам керувати різними пристроями за допомогою голосових команд через інтегровані пристрої, такі як Echo, Echo Dot та інші. Alexa підтримує широкий спектр пристроїв, включаючи освітлення, термостати, камери безпеки та побутову техніку.

Прогноз економії електроенергії за допомогою впровадження ІОТ рішень в hPh за рік



4

Використання протоколів у розумних будинках



5

СКЛАДОВІ СИСТЕМИ: КОНТРОЛЕР, ДАТЧИК РУХУ, ДАТЧИК ОСВІТЛЕННЯ

NODE MCU ESP8266

BH1750

HC-SR501

HC-SR501 Pinout

www.TheEngineeringProjects.com

Аналогові входи
 Входи за замовченням
 Цифрові входи
 Порти мікроконтролера
 Інші
 Живлення
 Заземлення
 Внутрішній пін

6

СКЛАДОВІ СИСТЕМИ: ЛАМПА, LED СТРИЧКА, РЕЛЕ НАПРУГИ, АКУМУЛЯТОР

Philips Hue White A19



LED стрічка Grow Light USB 5V



NCR18650BF



1-канальне реле модуль



7

Схеми підключення живлення компонентів

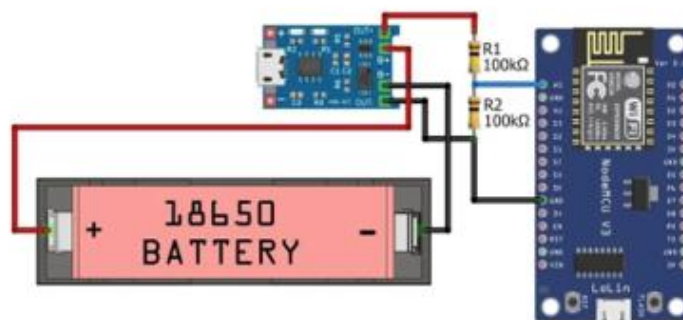


Підключення лампи до контролера через реле напруги для управління лампою



RobotChip

Підключення акумулятору до контролера через стабілізатор напруги

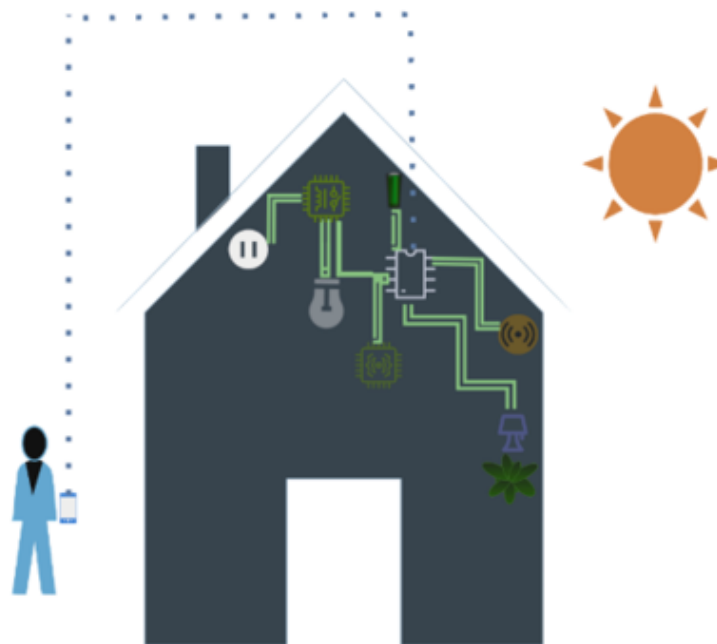


8

**СХЕМА ВЗАЄМОДІЇ КОМПОНЕНТІВ.ВЗАЄМОДІЯ
С КОРИСТУВАЧЕМ
ПРИКЛАД ДОСТАТНЬОГО ОСВІТЛЕННЯ**



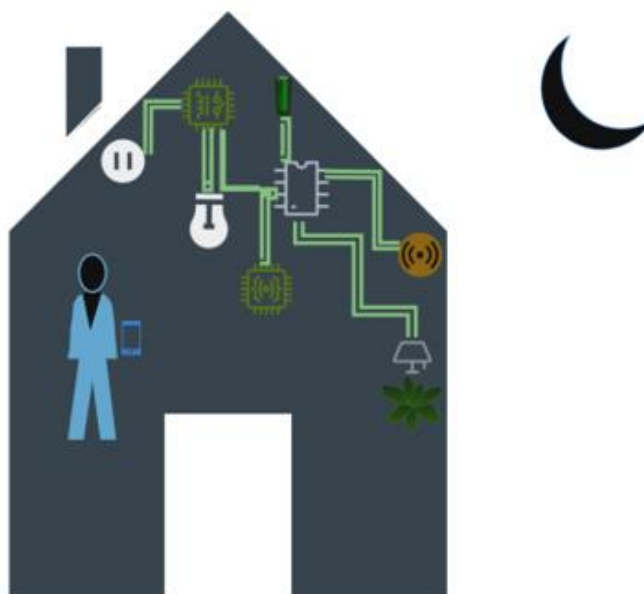
9



**СХЕМА ВЗАЄМОДІЇ КОМПОНЕНТІВ.ВЗАЄМОДІЯ
С КОРИСТУВАЧЕМ
ПРИКЛАД АКТИВАЦІЇ ДАТЧИКУ РУХУ**



10



**СХЕМА ВЗАЄМОДІЇ КОМПОНЕНТІВ.ВЗАЄМОДІЯ
С КОРИСТУВАЧЕМ
ПРИКЛАД НЕДОСТАТНЬОГО ОСВІТЛЕННЯ.
ВІДСУТНІСТЬ АКТИВАЦІЇ ДАТЧИКУ РУХУ**



11

▶ ВИСНОВКИ

- У світі останніх тенденцій до створення розумних будинків, автоматизація побутових процесів є важливим кроком на шляху до підвищення енергоефективності та зручності використання електроприладів. Використання IoT-систем для автоматичного управління освітленням, безпековими системами та іншими аспектами побуту дозволяє значно знизити споживання електроенергії та забезпечити комфортне проживання.
- Під час вивчення принципу роботи та використання розробленої системи автоматичного управління побутовими електроприладами були виявлені такі їх переваги: низька вартість, легкість програмування, висока інтеграція з іншими пристроями та платформами, гнучкість у налаштуванні та розширенні функціональності, можливість автономної роботи та віддаленого управління.
- Система автоматичного управління побутовими електроприладами на основі NodeMCU ESP8266 дозволяє інтегрувати різноманітні пристрої, такі як датчик освітленості та руху, розумні лампи, світлодіодні стрічки та реле, що забезпечує гнучкість у створенні сценаріїв автоматизації та підвищення комфорту в побуті.
- Однією з основних переваг системи є можливість віддаленого керування через веб-інтерфейс або мобільний додаток, що забезпечує зручність використання та контроль за всіма аспектами розумного будинку з будь-якої точки світу.
- Система автоматичного управління побутовими електроприладами може використовуватися для догляду за кімнатними рослинами, забезпечуючи оптимальні умови для їх росту та розвитку, а також для покращення якості освітнього процесу в університетах, надаючи студентам можливість здобути практичні навички з розробки та роботи з сучасними технологіями IoT.

12



ДЯКУЮ
ЗА УВАГУ